DOCUMENT RESUME

ED 031 281 LI 001 647

By-Veaner, Allen B., Ed.; Fasana, Paul J., Ed.

Stanford Conference on Collaborative Library Systems Development. Proceedings of a Conference Held at Stanford University Libraries, October 4-5, 1968.

Stanford Univ., Calif. Libraries.

Pub Date 69
Grant-OEG-1-7-077145-4428
Note-234p.
Available from-Office of the Financial Manager, Stanford University Libraries, Stanford, California 94305 ($7.00)
EDRS Price MF-$1.00 HC-$11.80
Descriptors-*Automation, *Cooperative Programs, *Information Systems, *Library Networks, *University Libraries
Identifiers-CLSD, *Collaborative Library Systems Development

The conference was convened (1) to disseminate information on the development of Stanford's library automation project, and (2) to disseminate information on the several and joint library automation activities of Chicago, Columbia, and Stanford, and (3) to promote heated discussion and active exchange of ideas and problems between librarians, university administrators, computer center managers, systems analysts, computer scientists, and information scientists. These conference papers present information on library automation projects at various universities and specialized information about institutions involved in bibliographic data processing activities. Topics specifically include: the Collaborative Library Systems Development (CLSD), the National Libraries Automation Task Force, the Biomedical Communications Network at the National Library of Medicine, the Book Processing System at the University of Chicago, the application of hardware and software in libraries, data link network and display terminals at Stanford, and other automation projects at Chicago, Columbia, and Stanford. (Author/RM)

ED031281

LI 001047

Stanford Conference on Collaborative Library Systems Development

PROCEEDINGS

of a Conference Held at Stanford University Libraries

October 4-5, 1968

Edited by
Allen B. Veaner
&
Paul J. Fasana

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
OFFICE OF EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL OFFICE OF EDUCATION POSITION OR POLICY.

Stanford University Libraries
Stanford, California 94305
1969

LI 001047

Stanford Conference on Collaborative Library Systems Development

PROCEEDINGS

of a Conference Held at Stanford University Libraries

October 4-5, 1968

Edited by
Allen B. Veaner
&
Paul J. Fasana

The papers of this Conference were supported by Office of Education Grant OEG-1-7-077145-4428. They express solely the opinions and professional judgment of their authors. Points of view or opinions stated do not, therefore, necessarily represent official Office of Education position or policy.

Publication of these *Proceedings* is a Stanford endeavor and not a formal part of the grant.

Stanford University Libraries
Stanford, Ca. 94305
1969

Acknowledgments

The editors wish to thank the following persons for their invaluable assistance in setting up the Conference and facilitating publication of the Proceedings: Mrs. Lois Butts, Conference Coordinator at Stanford; Miss Adriane Baron, former Secretary to the Automation Division, for her outstanding work in executing the details of conference arrangements; Mr. Tom Masterson, Head of the Stanford Photographic Service, for providing projection equipment and for tape recording the proceedings; Mr. Ken Watson, Head of the Meyer Library's Audio-Visual Service, for providing backup tapes; Miss Diane Clark, a Stanford student, for her patient, thorough job of transcribing the entire conference from the tapes; Miss Charla Meyer, Secretary to the Automation Division, for overseeing production of the final manuscript.

It was extremely gratifying to observe the enthusiastic participation of the invited speakers and the attendees. To them a special debt is owed.

Allen B. Veaner
Paul J. Fasana
April, 1969

TABLE OF CONTENTS

Page

Introduction 4

Conference Program 5

List of Registrants 6

The University of Chicago Library Automation Project, a Summary, by Herman H. Fussler 10

Automation Efforts at the Columbia University Libraries, a Summary by Paul J. Fasana 19

Stanford University Libraries, Project BALLOTS, a Summary by Allen B. Veaner 42

Collaborative Library Systems Development (CLSD) by Dr. Richard H. Logsdon 50

National Collaboration and the National Libraries Automation Task Force: A Course Toward Compatibility, by Samuel Lazerow 55

Management of the Design and Development of the Biomedical Communication Network, by Ralph A. Simmons 75

The University of Chicago's Book Processing System, by Charles Payne and Kennie Hecht 119

Economic and Operating Realities of Present Day Hardware and Software in Library Application, by W.F. Miller & R. Bielsker 144

Stanford's Data Link Network and Display Terminals, by R.M. Fredrickson and M.D. Lieberman 171

Computer Operating Systems and Programming Languages: A Critical Review of Their Features for Processing Bibliographic Text, by T.K. Burgess 192

Developing a Campus Based Information Retrieval System, by Edwin B. Parker 213

INTRODUCTION

The Stanford Conference on Collaborative Library Systems Development was convened for several purposes: (1) To disseminate information on the development of Stanford's library automation project supported by an Office of Education grant, and (2) to disseminate information on the several and joint library automation activities of Chicago, Columbia, and Stanford, and (3) to promote heated discussion and active exchange of ideas and problems between librarians, university administrators, computer center managers, systems analysts, computer scientists, and information scientists. To carry out the third objective effectively, the invitations were strictly limited to a small number of institutions known to be experienced in a wide range of bibliographic data processing activities. The animated discussions following the papers testify to the effectiveness of this procedure.

Papers given at the Conference were decisively oriented towards lending an air of technical practicality and economic reality to library automation, an endeavor which at times has lacked one or both of these qualities. The papers and discussions are published with the view of provoking enlarged discussions elsewhere, and the editors will welcome comments and critiques from readers.

R.D. Rogers
Director of Libraries

STANFORD CONFERENCE ON COLLABORATIVE LIBRARY SYSTEMS DEVELOPMENT

FORUM ROOM, MEYER LIBRARY, STANFORD UNIVERSITY, STANFORD, CALIFORNIA
OCTOBER 4-5, 1968

PROGRAM

Friday, October 4, 1968

9:00 - Registration and Coffee
9:15 - Welcome by Rutherford D. Rogers, Director of Stanford University Libraries
9:30 - Project Summaries:
    Chicago: Dr. Herman H. Fussler, Director, The University of Chicago Library
    Columbia: Mr. Paul J. Fasana, Assistant to the Director, Columbia University Libraries
    Stanford: Mr. Allen B. Veaner, Assistant Director of University Libraries
10:30 - Collaborative Library Systems Development (CLSD)
    Dr. Richard H. Logsdon, Director, Columbia University Libraries
10:50 - Coffee Break
11:00 - National Collaboration and the National Libraries Task Force: A Course Toward Compatibility, by Samuel Lazerow, Chairman, National Libraries Task Force on Automation
12:00 - Lunch
2:00 - Management of the Design and Development of the Biomedical Communications Network, by Ralph A. Simmons, Head, Information and Computer Sciences, National Library of Medicine, Bethesda
3:00 - Coffee Break
3:15 - The University of Chicago's Book Processing System, by Charles Payne, System Development Librarian, The University of Chicago Library; Mrs. Kennie Hecht, Programmer

Saturday, October 5, 1968

9:00 - Economic and Operating Realities of Present Day Hardware and Software in Library Applications, by Professor William F. Miller, Associate Provost for Computing and Professor of Computer Science, Stanford University, and Mr. Richard Bielsker, Computer Programming Manager, BALLOTS/SPIRES, Stanford University
10:15 - Coffee Break
10:45 - Stanford's Data Link Network and Display Terminals - What They Mean to the University's Information Retrieval Projects, by Mr. Roderic M. Fredrickson, Associate Director, and Mr. Mark D. Lieberman, Assistant Director, Campus Facility, Stanford Computation Center.
12:00 - Lunch
1:45 - Computer Operating Systems and Programming Languages: A Critical Review of Their Features and Limitations for Processing Bibliographic Text, by Mr. Thomas K. Burgess, Manager, Systems Development, Washington State University
2:45 - Coffee Break
3:00 - Developing a Campus Based Information Retrieval System, by Professor Edwin B. Parker, Department of Communication, Stanford University

LIST OF REGISTRANTS

STANFORD CONFERENCE ON COLLABORATIVE LIBRARY SYSTEMS DEVELOPMENT

FORUM ROOM, MEYER LIBRARY

STANFORD UNIVERSITY, STANFORD, CALIFORNIA

OCTOBER 4-5, 1968

Dr. Burton Adkinson
Head, Office of Science Information Service
National Science Foundation
Washington, D.C.

Mr. Richard Bielsker
Programming Manager
Institute for Communication Research
Stanford University

Mr. Donald Bosseau
Head, Systems Department
University of California
San Diego, California

Mr. Thomas K. Burgess
Manager, Systems Development
Washington State University Library
Pullman, Washington

Mrs. Rita Campbell
Archivist, Hoover Institution
Stanford University

Mr. Kenneth D. Creighton
Controller
Stanford University

Mr. Frank Kurt Cylke
Acting Head, Library and Information
Science Branch
Bureau of Research
Office of Education
400 Maryland Avenue, S.W.
Washington, D.C.

Miss Diana DeLanoy
Senior Systems Analyst
Stanford University Libraries

Professor William S. Dix
Director of Libraries
Princeton University
Princeton, New Jersey

Mr. Paul J. Fasana
Assistant to the Director
Systems Division
Columbia University Libraries
Butler Library
New York, N.Y. 10027

Mr. Roderic M. Fredrickson
Associate Director
Campus Facility
Stanford Computation Center
Stanford University

Dr. Herman H. Fussler
Director
The University of Chicago Library
Chicago, Ill. 60637

Miss Susan Geddes
Senior Systems Analyst
Stanford University

Mr. Robert Golter
Assistant Director and Librarian
of J. Henry Meyer Library
Stanford University

Mr. John Gwynn
Senior Systems Programmer
Project INFO
Stanford University

Mr. Donald P. Hammer
Head, Library Systems Development
Purdue University Libraries
Lafayette, Ind. 47907

Mrs. Kennie Hecht
Programmer
The University of Chicago Library
Chicago, Ill. 60637

Mr. J. Myron Jacobstein
Director
Stanford Law Library

Mr. Frederick Kilgour
Director
Ohio College Library Center
1858 Neil Avenue
Columbus, Ohio 43210

Mr. William R. Kincheloe
Senior Research Associate
Stanford Electronics Laboratories

Professor Kennth King
Director, Computation Center
Columbia University
New York, N.Y. 10027

Mr. Samuel Lazerow
Chairman, U.S. National
Libraries Automation
Task Force
Library of Congress
Washington, D.C. 20540

Mr. Ed Leys
Planning Office
Stanford University

Mr. Mark Lieberman
Assistant Director
Campus Facility
Stanford Computation Center

Dr. Richard Logsdon
Director
Columbia University Libraries
Butler Library
New York, New York 10027

Mr. R. W. MacDonald
Systems Analyst
University of British Columbia
Vancouver, B.C., Canada

Mr. Colin MacLeod
Project INFO
Stanford University

Dr. Stephen A. McCarthy
Executive Director
Association of Research Libraries
1527 New Hampshire Avenue, N.W.
Washington, D.C.

Mr. Rob McGee
The University of Chicago Library
Chicago, Ill. 60637

Mr. Colin McKirdy
Acting Head, Data Processing
Department
Harvard College Library
Cambridge, Mass. 02138

Mr. Paul Miles
Assistant University Librarian
University of California
Los Angeles, California

Professor W.F. Miller
Associate Provost for Computing and
Professor of Computer Science
Stanford University

Mr. Peter Moncrieff
Programmer
Columbia University Libraries
New York, N.Y. 10027

Mrs. Eleanor Montague
Systems Librarian
Stanford University Libraries

Mr. Wallace C. Olsen
EDUCOM
5805 Marbury Road
Bethesda, Maryland

Dr. Edwin B. Parker
Associate Professor, Department
of Communication
Stanford University

Mr. Charles Payne
Head, Systems Development
The University of Chicago Library
Chicago, Ill.

Mr. Paul Reimers
Director, Information Systems Office
Library of Congress
Washington, D.C.

Mr. Rutherford D. Rogers
Director of Libraries
Stanford University

Mr. Paul St. Pierre
Chief, Systems Analysis and Data
Processing Office
The New York Public Library
Fifth Avenue and 42nd Street
New York, New York 10018

Dr. Russell Shank
President, Information Science
and Automation Division
American Library Association
c/o Smithsonian Institution
Washington, D.C.

Mr. Ralph Shoffner
Head, Operations Task Force
Institute of Library Research
University of California
Berkeley, Ca.

Mr. Ralph A. Simmons
Head, Information & Computer
Sciences Staff
National Library of Medicine
Bethesda, Maryland

Dr. James E. Skipper
Director of Libraries
University of California
Berkeley, Ca.

Mr. Donald Smith
Head, Serials Department
University of Toronto Libraries
Toronto, Ontario, Canada

Mr. Carl Spaulding, Specialist for
Automated Systems
Council on Library Resources
1028 Connecticut Avenue, N.W.
Washington, D.C.

Dr. Richard F. Staar
Associate Director Designate
Hoover Institution
Stanford University

Mr. James Thompson
Systems Department
University of Michigan Library
Ann Arbor, Michigan

Mrs. Susan Watson
Head, Operations Planning Office
University of Pennsylvania
Philadelphia, Pennsylvania

Mr. David Weisbrod
Director, Systems Development
Yale University Library
New Haven, Conn.

Mr. William Welsh
Director, Processing Department
Library of Congress
Washington, D.C.

Mr. Allen B. Veaner
Assistant Director of University
Libraries for Automation
Stanford University

Mr. David C. Weber
Associate Director of Libraries
Stanford University

Mr. Jerrold West
Programmer/Analyst
Stanford University Libraries

Mr. Charles H. Stevens
Staff Member
Project INTREX
M.I.T.
77 Massachusetts Ave.
Cambridge, Mass. 02139

The University of Chicago
Library Automation Project
A Summary

Herman H. Fussler
Director
The University of Chicago Library

A Paper Prepared for the Stanford Conference on Collaborative Library System Development, October 4-5, 1968

Rogers: Herman Fussler was born in Philadelphia in 1914. He received his undergraduate degree at North Carolina and his Ph.D. in Library Science at Chicago. He began his library career at the New York Public Library in 1936, but soon was called to the University of Chicago where he was successively Head of the Department of Photographic Reproduction, Science Librarian Assistant and Associate Director of University Libraries before assuming the director ship in 1948. From 1942 to 1945 he was the Assistant Director of the Information Division and Librarian of the Manhattan Project in Chicago. He has atteneed several international conferences on documentation, has served as member and Chairman of the Board of Directors of the Center for Research Libraries, is a member of the Library Advisory Board of the Air Force Academy, and from 1963 to 1967 he was Regent of the National Library of Medicine. He was one of the few librarians to sit on the National Advisory Commission on Libraries.

* * * * * * * * * * *

Since a more detailed report on the University of Chicago Library Automation project is scheduled for later in the day, I would like to use a portion of my time to try to relate the Chicago project, in a very brief and general way, to some of the more general forces that seem likely to affect large research libraries in the foreseeable future and that have affected much of our project's planning and concepts.

At the present time it can be argued that there are at least two braod, closely interrelated, and somewhat overlapping elements that are distinguishable in the communication processes in which large, research libraries are a critical element: [1] those processes having to do with content and bibliographical analysis and control that serve to alert a reader to the existence, degree of relevance, and location of pertinent material of information; and [2] all those processes having to do with collection building, organization of collections for use, and related public services--in short, all the operations related to the provision of textual and document access for readers.

Obviously these processes at times are so intimately interrelated that they are hardly distinguishable from one another, and there are observers of the current scene, as we all know, who argue that the distinction in these processes, if there is one, will soon disappear entirely in the world of the real-time, console-based, reader-computer dialogue with massive storage, search and display capabilities. Here the act of specifying the query will supply the needed answers. Fortunately or unfortunately, there are some technical, economic, and intellectual reasons for believing that while this completely integrated approach is now possible, at fairly high costs, for very limited, fact-oriented bodies of literature, it is not likely to occur soon, and perhaps never, for the vast corpus of recorded knowledge and information that is the common concern of large research libraries. Nonetheless, the rate of change in research library concepts and operations can and must be much greater in the future that it has been in the past if we are to meet visible educational and research access needs satisfactorily.

It now seems reasonable to assume that much of the basic intellectual work of bibliographical control and analysis of the content of research materials will increasingly be generated by the national libraries, by other Federal or international agencies, by professional societies, or others, and that proportionately less will need to be undertaken locally except in areas of exceptional interest and competence. It also seems reasonable to anticipate very significant improvements in the quality, speed, and scope of bibliographical and content control of research materials in, say, the next ten years or so.

These improvements in the availability and the quality of bibliographical control can only increase the already heavy pressures for improved physical, textual, or document access--as these processes have variously and somewhat unfortunately, come to be called. It is not necessary in this company to attempt a description of the current state of physical access in the typical large research library, for I assume most of you know and would agree that is is unsatisfactory to a significant percentage of the users of libraries.* It may, however, be in order to try to identify at least a few of the underlying problems, issues, and principles that may be particularly relevant to efforts to improve physical access. One fundamental problem, curiously overlooked by many observers, is that the research library must serve both highly expansive and open-ended needs for resource access and related services with, at any point in time, quite finite resources in staff, money, space, etc. The rate of expansion in demands, though not subject to satisfactory quantitative measurement--a serious problem in itself, has, in general, clearly been more rapid than that for library support. The latter, in turn, has been sufficiently rapid to be a source of university administrative concern. Since the benefits of library resources and services are very difficult to measure, the determination of appropriate levels of library support and the optimum allocation of the available resources present difficult problems.

These determinations presently are heavily dependent upon tradition, intuition, inter-institutional comparison, and a variety of similar devices. This is all perhaps a very lengthy way of saying that university libraries must look more critically and systematically at cost/benefit analyses of different patterns of resource allocation as well as the quality and scope of the services to be provided.

It is my assumption that improved inter-institutional patterns must be found for some major segments of collection development and that individual libraries must greatly improve the processes for local physical access to all kinds of resources. To grossly oversimplify the nature of the problem, putting one's hands on a needed book or reference must be made easier and faster, and more certain.

The Chicago automation effort has been conceived as one of the needed responses to this broad class of access problems, including the utilization and generation where necessary, of the necessary bibliographical control data. The project has been focused primarily upon the data processing operations of a relatively large research library. These processes are exceptionally complex and are critically related to the response capabilities of any research library. Many of the data-processing operations of a library are highly structured and follow quite formal rules; they are thus, in theory, especially susceptible to computer-aided data processing.

---

* The phrase "large, research library," in itself, of course, excludes the truly massive problems of institutional and regional inequities in physical access to research resources. I should also emphasize that I am using physical access in a special way to include local cataloging as well as other bibliographical work, based upon LC or other external inputs.

Within the framework of these general concepts, the specific objectives of the system under development oy the University of Chicago have included the following: [1] to improve in a significant way the response times of a large library in most of its basic routines related to data processing, i.e., acquisitions, cataloging, circulation routines book status data, etc., with the primary aim of improved service to readers; [2] to build a data base upon which new or improved services to readers might be built at relatively low incremental costs : [3] to assemble better and much more current library performance and operating data than are available with manual systems; [4] to stabilize, and possibly in some cases to reduce, the unit costs for many library routines; [5] to provide library systems capable of relatively easier evolution and adaptation to meet changing or more sophisticated reader or institutional requirements: [6] to build library data-handling systems that will be able to utilize externally-generated bibliographical data swiftly and efficiently; [7] to provide systems that can respond effectively to sharp, seasonal load changes as well as to long-range load increases without proportionate changes in staffing requirements; and [8] to provide systems that will respond more adequately to certain other kinds of staffing problems in connection with routine, data processing operations.

Among the conditions or requirements imposed upon the basic system designs were the following: [1] the system should be based upon the levels of bibliographical analysis and control for monographs and serial titles that are now in general use, with an evolutionary capability for handling more sophisticated levels of control and analysis in the future; [2] the system should be one that in cost and performance could be justified and supported by regular University funds once it was fully operational; [3] the initial system design was to be based upon the use of a common data base and common software, wherever appropriate, rather than upon existing departmental or other functional or administrative units; [4] the system or systems, whether mechanized or not, were to be as responsive as possible to functional or user's needs, in terms of data handling capacity, on-line/off-line processing and access, character-set size, format, legibility, costs, reliability, etc; [5] wherever possible the system should be based upon a single input of appropriate data, with the ability to update, extend, reformat, correct, or delete data to match the typically evolving and changing state of library processing information.

Since the date of the NSF grant--some 27 months ago--we might mention the following selected list of accomplishments, on most of which Charles Payne will provide details later.

1. The design of a bibliographic data handling system has been completed.
2. Documentation of the bibliographic data element descriptions and tagging code lists, with descriptions in detail of the input, editing and correction features, handling of call numbers, holdings statements and output distribution has been completed and publicly distributed. This effort has had a constructive influence, we believe, on the design of the LC MARC II system, which we expect to utilize as an integral part of the system soon after MARC II is operational.
3. An on-line computer system for batch and remote library terminal operation has been developed.
4. Programming to handle input, processing, and output of bibliographic data has been developed. Catalog card formatting and printing programs are operational on a high speed printer.
5. The data element description and processing requirements for the handling of acquisitions data have been developed, including development of a book fund coding system. Much of this work is now being reviewed with the CLSD group.
6. A library data processing unit has been set up for I/O operations.
7. Circulation charge cards and book pocket labels are being formatted and printed from the common data base on a high-speed line printer.
8. A large amount of work has been undertaken on circulation systems studies and design. Existing operations have been rigorously studied and requirements drawn up. Several equipment configurations have been critically examined. It is hoped to complete the design and testing of a machine-aided system, that initially would be off-line, by late this year. Such a system would be an interim system until such

time as suitable, multiple, computer terminals at reasonable cost, with adequte computer storage can be made available, Related studies have been made of I.D. card systems and production methods and other library requirements. I should emphasize that we regard good circulation systems as of critical importance in improving physical access. We also believe that highly responsive circulation systems for large libraries with very large files and I/O rates are much more difficult to design, within reasonable sets of requirements, than seems to be commonly assumed.

9. We have undertaken a variety of systematic studies to determine unit-costs and other performance data in order to have a somewhat better picture of the "before" and "after" situation.
10. During these processes we have upgraded the computer and peripheral equipment substantially. We started with an IBM 360/30 with 32k core memory, the BOS machine operating system--then the only one available--and machine assembly languages--also the only feasible language. We went to a 360/40 and DOS on an interim basis, and are now operating, and hope to stabilize for some time, on IBM's OS system and a 360/50.

The total staff used in 1967/68 came to approximately 15 F.T.E., roughly divided as one-half professional, including regular professional staff assigned on temporary, part-time bases to various segments of the project. The largest portion of the clerical staff was related to input/output data processing; if this portion of the staff were excluded, as essentially non-developmental, the manpower investment in the project last year was approximately 10 F.T.E., virtually all of which was professional. Approximately 7 F.T.E. went into programming and systems work. The NSF grant was for $452,000 for a 3-year project with substantial additional matching funds from the University. The NSF funds have been used primarily for computer costs and programming work.

Let me conclude with a few general observations:

1. The memory requirements for shared-time, for remote terminals, for a reasonable array of peripheral equipment, and for reasonable bibliographical operational requirements are quite large, yet relatively little computational use is made of the computer. This argues we believe, for shared-time use of moderately large to large computers in applications of this kind, given the present stage of computer

cost. Batch operations of independently designed routines can probably be programmed more easily and run on smaller computers. Their evolutionary capability is likely to be at a significantly lower level.

2. Library processes in general and bibliographical data processing in particular in a large research library environment present more difficult and more complex problems than librarians, systems analysts, or computer specialists and programmers normally anticipate. Estimates of the costs for developing software are therefore extremely difficult to make, and these estimates tend--as is widely know--to be low.
3. We are still persuaded that good programs with the right computer and with appropriate peripheral equipment appear likely to be quite powerful aids to effective library operations.
4. The development and implementation of new automated systems, where they must intermesh with on-going, daily, operational needs, requires extremely careful planning for transitions. Even with such planning there are likely to be dislocations, delays, and staff frustrations. Ideally, of course, a library would operate new systems in parallel and completely separated from the existing manual systems they are to supersede, until all operational, mechanical, and software problems have been identified and solved, and full operational reliability of the new system has been thoroughly established. Unfortunately, this approach requires both staff, money, and schedule time that are rarely available.
5. Advance estimates of operating as well as developmental costs for new systems are difficult to project, and the relationship of these costs, adjusted for changes in effectiveness or performance, to existing operational or unit costs, also present difficult analytical problems. This problem can be particularly important and difficult in attempting to predict between on-line and off-line processing effectiveness and costs.
6. There is a conspicuous absence of certain kinds of badly needed peripheral equipment for library operations. The pursuit of information on possibly suitable equipment is time consuming and manufacturing responses to specialized needs, at reasonable costs, tend to be slow or entirely absent.

7. There is also an absence of certain badly needed general data management software packages to provide file organization, update, and retrieval capabilities desirable in library processing operations. Existing systems are considered prohibitively expensive in cost and core dedication requirements, and may demand total dedication of a time-shared machine for the data management activities.

Automation Efforts at the
Columbia University Libraries
A Summary

Paul J. Fasana
Assistant to the Director
Columbia University Libraries

A Paper Prepared for the Stanford Conference on Collaborative Library Systems Development, October 4-5, 1968

Rogers: Paul Fasana comes from Bingham Canyon, Utah. Paul was educated at the University of California in Berkeley, where he majored in language and literature and library science. He began his professional library career as a cataloger at the New York Public Library, from where he moved to Itek Laboratories as a systems engineer. From 1963 to 1964 he was Chief of Cataloging, U.S. Air Force Cambridge Research Laboratory Library, and from 1964 to 1966 he was Assistant Coordinator of Cataloging at Columbia University Library. In 1966 he became the Assistant to the Director of University Libraries for Automation. He has served on the Board of Directors of the Information Science and Automation Division of A.L.A. and has been Chairman of the Committee on Dissemination of Information. He has written extensively on automation and bibliographic control. In addition to his other duties, he is now secretary of the Collaborative Library System Development Program.

* * * * * * * * * *

INTRODUCTION

The purpose of this presentation is to describe automation and systems efforts at Columbia University Libraries. A casual glance at Columbia's efforts suggests a confused state. I hope that by the end of my presentation this confusion will be removed and that a controlling thread is revealed which relates individual projects.

In order to help you understand the total effort, I've attached a list of current and past projects (see list, page 31). As you can see from the list there are a variety of projects and it is not immediately apparent how they integrate, if at all, or what the overall structure is. We like to think that there is a atructure which allows individual projects to ultimately come together into an integrated or total system. At present this is more an objective than a reality, and we realize that this objective is several years in the future.

HISTORICAL BACKGROUND

Possibly the best way of explaining our attempts at Columbia is to give you an historical perspective. In 1964 after an overall review of existing manual operations, we formulated a conceptual approach to the problem of library automation. At this point, it is difficult to define precisely what this conceptual approach was. On one level, it might be described as an attitude allowing the Libraries to begin to understand computer technology and the range of problems to be explored with respect to library operations and computers. On another level, it did establish an environment wherein theories and ideas could be tested and, if proved valid, implemented into a real work situation. We realized then that it would be impractical to develop ideal systems and attempt to impose them on reality. Automating library operations requires a long period of transition during which successive computer-based systems can be designed and tested.

Roughly two years were spent looking at library operations. Our primary objective was to get a realistic overview of existing library operations. Our secondary objectives during this period were threefold: first, to assess the potential of computer and allied technologies with respect to library operations; second, to develop a plan for introducing electronic data processing into a traditional library environment; and third, to define what the role of the research library should be in this new electronic era.

I would like to mention four specific results of this preliminary study. First, we concluded that the state of the art of library automation in 1964 (and still to a large degree today for large library environments) was quite primitive. Although there had been several widely publicized library automation projects, little of the experience gained in these projects was directly pertinent, we felt, to a university library environment. At best, they revealed the depth of the librarian's ignorance with relation to automation and seemed to indicate that library automation should be considered essentially a research effort. Secondly, we concluded that the machines themselves were ill-suited to library operations and the effort required to adapt second generation computers to library operations created enormous problems. At the time, we stated that libraries need mass storage capability, random access to files, on-line enquiry, and programming languages to handle variable length character strings. We realize now, even though we are beginning to have some of these features with third generation computers, that the task of adapting computers to library operations is still a difficult problem and, in many ways, more subtle. Thirdly, we realized that the university environment was in a state of transition, and that the library would have to be sensitive to changes. Libraries in the past have played an essentially passive role, that of acquiring materials, storing them and making them available on request. The university library of the future will be required to play a decisively different, more aggressive role, if it wants to retain its primacy as the central information store or center on campus. It is still uncertain what this role will be precisely, but increasingly there are signs to indicate what direction the library should be going in. Information is being produced in many new forms; user groups are emerging having different and often conflicting requirements. Each of these factors greatly affects the nature of the university library. (I will have more to say about changing user requirements in a moment.)

Fourthly, the approach a university library adopted to automation was critical. There seemed at that time to be two essential approaches. At one extreme, there were those who insisted that efforts be directed towards developing a total, grandiose system encompassing the entire range of library operations. We decided that this approach was neither practical nor feasible. At the other extreme, there were those who insisted that a far more modest, piecemeal approach was needed. When viewed in a systems context, a large research library is essentially an aggregate of systems. We realized that initially the interface points between these sub-systems is minimal, but as momentum is gained, the interrelation of these sub-systems becomes critical. Therefore, extremely careful, detailed planning would be necessary if the ultimate objective was to create an efficient, integrated, total system. We rationalized our decision by stating that in using this approach, the librarian could gain experience gradually, allowing him to control the rate at which automation efforts progressed, and to choose those areas where automated techniques were most needed and could be used most effectively. This last point has proven to be especially important.

Over the years libraries have developed procedures and standards which are extremely restrictive and geared primarily to manual operations. Many of these manual procedures are totally unsuited to computer operations; many of the established standards are of dubious value in an automated system. Revision of procedures and standards is necessary but it cannot be done unilaterally or hastily. A research library has a responsibility to the general library community and an extremely large investment in its manual files. As a result of these two factors, we formulated an approach wherein we first defined an area for study, analyzed what was being done to identify essential functions being performed, and then translated these functions into computer capabilities. The resulting system is, as a consequence, generalized and applicable beyond the immediate or particular environment studied. We have to date successfully employed this approach in two areas within the Columbia Libraries: circulation and reserve book processing. This experience also provided us with the incentive to participate with Chicago and Stanford in the Collaborative Library Systems Development Project. It soon became apparent to us using this approach, that automated library systems would have to be designed and implemented in phases or successive generations, each being more refined than the preceding. This means that automation in a research library should be viewed as a long range effort, often times with no immediate hope of savings or success.

THE USER PERSPECTIVE TO LIBRARY AUTOMATION

I mentioned earlier that users have played an important role in defining Columbia's approach to automation. The process of becoming aware of the users' role has been gradual and subtle and is still not completely understood. At this point we know that there are at least three major types of user, each requiring different kinds of service. The services required by these users are at times seemingly incompatible, but we feel that eventually the Libraries can develop a single or integrated system which will satisfy their different requirements. We do not feel that developing separate systems for each of these groups would be in the best interests of the Libraries, although it would be a far simpler task at present. In an attempt to cope with this problem, we have coordinated all library systems efforts into a single office thereby pooling technical personnel and using them in several library projects.

The three user groups that we have identified are as follows: the student population; the research groups; and the librarians. I would like to spend several minutes describing generally how each of our projects relates to one or another of these user groups.

The Student Group:

A university library has a strong commitment to the instructional process. In the past twenty years, the nature of this commitment has changed radically; university libraries have in general been insensitive to these changes in not assessing the kind or amount of service they should provide. We were acutely aware of this at Columbia in two major areas: reserve book processing and book circulation.

We discovered in Reserve Book processing that we were literally moving tens of thousands of books each semester and creating an equal number of records. The amount of effort and money expended throughout the library system was incalculable; more importantly, the service provided was, at best, partially effective. In terms of actual numbers, we estimated that in one department library we processed in a typical semester 400 to 500 course reading lists; a typical list averaged 50 to 60 titles; the average number of volumes per title was 5 to 10; and the number of records needed to control and transfer each volume was 3 to 4. When multiplied out, the figures are astronomical. An intensive analysis of several reserve environments was conducted which resulted in the design of a computer based reserve processing system. The system is at present being implemented in two of the Libraries' largest reserve environments. The system as designed creates a variety of lists which are used for processing, public reference, and professor notification. It also produces machine readable inventory cards which assist the librarian in physically processing books onto and off of reserve. The present system is essentially a batch oriented system, although on-line terminals are used to input data. A later phase is planned which will make greater use of on-line, conversational processing and will incorporate certain circulation functions.

In the area of regular book circulation, we have designed and implemented a batch oriented system which is in production use in three large libraries. Although the system took less than three months to design and program, it required almost a year to successfully implement. It was our first major effort which in part explains why implementation was so lengthy. When we began, we were quite innocent of the problems of implementation and have, as a result of this project, come to the tentative conclusion that the major problem in a library automation project is not technical but personnel. The system has proved to be moderately successful in circulation environments having widely varying loads (1,000,000 plus in Central Circulation to 100,000 plus in the Business Library). The machine system costs roughly 10% more than the manual system, but has several advantages over it, such as greater growth potential, greater flexibility, and more accurate and up-to-date files (file size at present is approximately 100,000 records). In the near future, a revised design will be tested to incorporate source data collection procedures which will decrease the amount of input processing time significantly. In the revised system, machine sensible bar codes and optical scanning will be used to capture patron ID and charge data.

Within the next year or so, an on-line system integrating reserve and circulation functions will be developed and tested. Once this has been successfully implemented, we will begin to work to interface circulation and reserve procedures with cataloging.

The Research Group:

Within the past twenty years, universities have increasingly assumed responsibility for basic scientific research which in turn has created a new research community requiring more and better service than university libraries have been willing or able to provide in the past. From all indications this trend will continue and grow; university libraries are more and more being forced to acknowledge the demands of this group and react responsibly to their needs. A basic decision in this area is whether the libraries should attempt to integrate these specialized needs with more conventional library procedures, or develop services and systems tailored especially for the needs of the group. Many argue that the problem of libraries and computers is in itself an enormous task and that libraries should concern themselves with developing systems which allow them to do traditional operations first. Others, and Columbia is among this group, feel strongly that it is both desirable and technically feasible to integrate these requirements into a single effort. Further, that the interchange that is possible between such diverse efforts is mutually beneficial.

At Columbia the Libraries System Office is responsible for the technical development of two specialized data centers, both of which use computers extensively. Research work done in each of these data centers has afforded the Systems Office the possibility of testing new and innovative approaches to the organization of materials, the analysis of information, and the retrieval of information. I would like to spend several minutes describing several aspects of these projects and show how they have contributed to our overall effort.

With support from the National Science Foundation, the Lamont Geological Laboratory maintains an information center for the International Upper Mantle Project (called the World Data Center for Research on the Upper Mantle). This center is responsible for acquiring, analyzing, and disseminating research information on the Project from the entire world. During the past three years, the center has acquired material at the rate of several hundred reports a year in every major language. In 1967 the Center published a book catalog of its holdings using computers and tab equipment.

The response to the catalog was extremely favorable leading the Center to the conclusion that a regular dissemination service should be developed. After detailed analysis, the Systems Office decided that every attempt should be made to design a system which would take advantage of the standards work done by the Library of Congress MARC Project. Accordingly, the MARC II format was adapted to the type of data used in the Upper Mantle Center and a MARC compatible encoding format was developed. During the past six months, input encoding procedures have been designed and the entire Upper Mantle data file has been converted. At present programs are being tested which accept encoded data to produce a book form catalog having a classified section supplemented by author and permuted title indices. Before the end of the year, a comprehensive book catalog of the Upper Mantle's entire holdings will be published. Developing the programs for this project afforded the Systems Office the opportunity of experimenting and gaining experience with inputting data with an extended character set, in manipulating variable length character strings within the computer, and in testing the adaptability of several programming languages for text manipulation. The results have been so favorable that schedules have been established to use these same procedures and programs in the Parkinson Information Center, in the main Libraries' book cataloging, and in cataloging for special collections.

The Parkinson Information Center, supported by a grant from the National Institute of Neurological Diseases and Blindness, has been in operation for more than four years and has acquired, analyzed, and stored more than twenty thousand citations dealing with Parkinsonism and related disorders. The Parkinson Center is one of five centers, each of which deals with a particular aspect of the nervous system; eventually it is planned that these five centers will be wire linked to provide depth search capability to support the National Library of Medicine's Medlars service. The system as it is presently designed uses computers to encode bibliographic data, to maintain a thesaurus of terms, to produce a bi-weekly announcement bulletin, to provide an SDI service for medical personnel, and to do subject searches. Since the system was designed and implemented several years ago, it is judged at present to be outdated and archaic. Plans have been developed to revise certain aspects of the system. The first aspect of the PIC system to be revised will be procedures for encoding data. The Upper Mantle system mentioned earlier will replace PIC's input procedures. Accordingly, a computer program is being written to convert records in the PIC machine files to the MARC compatible format. It is anticipated that by early 1969, the entire PIC system will be re-designed and processed on the central IBM 360 computer.

A major emphasis in the PIC project has been to develop depth indexing techniques and tools. During the past three years, a highly structured thesaurus of terms has been developed which is complete in itself, and also nested or compatible with the National Library of Medicine's MESH list. The machine algorithms used to up-date the thesaurus have been studied and found to be applicable to the maintenance of traditional authority lists. Many of the specialized products and services provided in the PIC system have been analyzed and evaluated in terms of their possible application in a traditionally oriented environment. The experience has proven to be extremely useful and certain aspects, such as periodic announcement bulletins and machine search strategies, are already being incorporated into our design of an acquisitions and cataloging system.

The Librarian as a Specialized User Group:

Early in our work we realized that library computer systems must eventually be taken over and run by librarians. Therefore, the needs of the librarian must be considered in much the same way as other user groups. The librarian's needs are in many ways more subtle and complex, in that the librarian not only uses the system, but must also be responsible for monitoring the system. Library procedures have developed over a long period of time and are to a large degree controlled by standards outside a particular library. As a consequence, operations (i.e., acquisitions, cataloging, etc.) in different libraries tend to be quite similar; likewise, problems in different libraries can be thought of as being basically similar. Differences usually exist only on a procedural level. With this premise in mind we decided that any development work undertaken should to the largest degree possible be general, having applicability beyond our particular environment. We had discovered in two of our smaller efforts that it was both feasible and practicable to generalize about the functions performed in an activity and to design generalized systems based on the essential functions performed. The systems so developed, we found, could with a minimum of modification be used in different environments. Having demonstrated that this could be done internally, we wanted to test this approach on a larger scale. Since the analysis and design effort required in any library automation project is great, we saw a second possible benefit, that of being able to collaborate with other library automation efforts to reduce the cost and effort of developing major bibliographic systems. It was at this point that we became involved with Stanford and Chicago and the idea of the Collaborative Library Systems Design Project took shape. Another presentation will describe in detail the objectives and accomplishments of this project; I would simply state at this point that, even though we have had only six months experience, and in spite of the problems of distance, terminology, and hardware differences, a great deal of valuable collaborative work has been accomplished.

For the past year we have been devoting considerable time and attention to the Libraries' central processing system, that of acquisitions and cataloging. As anyone who has worked in a large library realizes, these activities are extremely complex and cumbersome. The flood of printed materials during the past two decades is seriously threatening the ability of any manual system to cope with it. It seems that, if the library is to survive, it must radically revise its procedures to make use of computer technology. In an on-going operation where there is a great investment in files and personnel, this is an extremely difficult task. If the problem was restricted to files and records, the task would be essentially technical and solutions would be more readily achieved. But the problem involves the librarian who must participate in the design of any new system because he alone understands the subtleties and vagaries of the existing system; and in a new system, the librarian will have to assume the responsibility of running the system.

Keeping all of these factors in mind, we have tried in analyzing acquisitions procedures to have a fresh, innovative attitude towards the design of a new system. For example, it became obvious that one of the major annoyances in acquisitions centered around fiscal responsibilities. After detailed study, we concluded that it would be impossible to design an efficient system around invoice processing. Therefore, we have been exploring the possibilities of using blank checks as order forms. While exploring the ramifications of this possibility, it became apparent that it might be more efficient to have the Libraries assume responsibility for the entire fiscal process, including check-writing, encumbering, and bookkeeping. At present, the system design incorporates all of these features. In the area of acquisitions process control, we have been studying the points of interface between the librarian, materials, and the computer system, and trying to establish what is, in fact, the proper combination of on-line and off-line processing. The popular thought is that on-line processing is preferable across the board. We feel that this is not necessarily the case and that there are certain operations which are more conveniently done off-line. For example, certain searching activities can be done more conveniently and easily against printed lists rather than through terminal enquiry. In all of these considerations, we are guided by the experience and need of the librarians themselves, rather than by the whims of computer experts.

CONCLUSION

What I have tried to suggest in this presentation is that, though there are a number of projects in progress at Columbia seemingly unrelated, they do, in fact, interrelate. It is difficult at times to keep firm control over all of these projects, and there is always the threat that the individual parts will not mesh. In spite of this, we feel that the benefits to be derived from exposing librarians and computer experts to library problems and allowing them to work together in a dynamic, quasi-research environment more than offset the possible dangers.

COLUMBIA UNIVERSITY LIBRARIES

INVENTORY OF COMPUTER BASED AND SYSTEMS ORIENTED PROJECTS

Past Projects:

a. 1963. Simulation of Columbia University Library (SCUL). A computer simulation model was developed to research library activities. Partially supported by a grant from the Council on Library Resources.
Status: Preliminary study completed; project discontinued for lack of funds.

b. 1964-65. Library Systems Study. A study of the Library's total operations was conducted by a team made up of library staff and IBM researchers. A total processing system was designed making extensive use of computers in an on-line environment.
Status: Conceptual Design completed.

c. 1961-66. Columbia-Harvard-Yale Medical Computerization Project. A cooperative effort to develop automated techniques for acquisitions, and cataloging. The final system was conceived of as an on-line, wire linked information network. Partially supported by a grant from the National Science Foundation.
Status: Discontinued.

d. 1967-68. Acquisition System Study. Processing functions for acquisition studied and described.
Status: Discontinued. (See below,g in Current Projects)

Current Projects:

a. 1964-65. Cost Analysis Study. A cost analysis was done of selection, ordering, and cataloging functions for science monographs.
Status: Initial study completed; unit cost for operations established.

b. 1964-present. Parkinson Information Center. A project to design, and operate a computer-based information center to collect, organize, and disseminate information in the subject area of Parkinsonism and related disorders. Work done under contract for the National Institute of Neurological Diseases and Blindness.
Status: Production mode for input processing (IBM 1410); Production mode for thesaurus maintenance (IBM 1410); testing/production mode for computer searching (IBM 7094); re-design for IBM 360 75/91 in progress.

c. 1965-present. Upper Mantle Project (IGY). A project to design, and operate an information center to collect, or-

ganize, and disseminate on a world-wide basis data of the Upper Mantle Project. Computer used to create book form catalogs. Partially supported with funds from the National Science Foundation.
Status: Production mode for input processing; production mode for book catalog production (IBM 360 50/75).

d. 1966-present. Union List of Serials. A project to create a union list of serials for engineering, science, and medicine (upwards to 10,000 titles) using the computer for reformatting, and listing purposes. Conceived as the first phase of a projected serials automation project. Final system will use a computer in an on-line, real-time mode for ordering, check-in, cataloging, and binding.
Status: Input data for union list complete; format programs written and tested (IBM 360 50/75).

e. 1966-present. Circulation. A computer-based circulation system was designed and programmed, and has been tested in several environments having varying work loads.
Status: Fully operational in Central Circulation and Burgess-Carpenter Library; partially implemented in Business Library.

f. 1967-present. Reserve Book Processing. A computer-based reserve system was designed and programmed. The system focusses on record creation, file management, and book inventory aspects of reserve processing. Partially supported by a grant from the U.S. Office of Education.
Status: Programs written and tested; parallel implementation in College Library in progress.

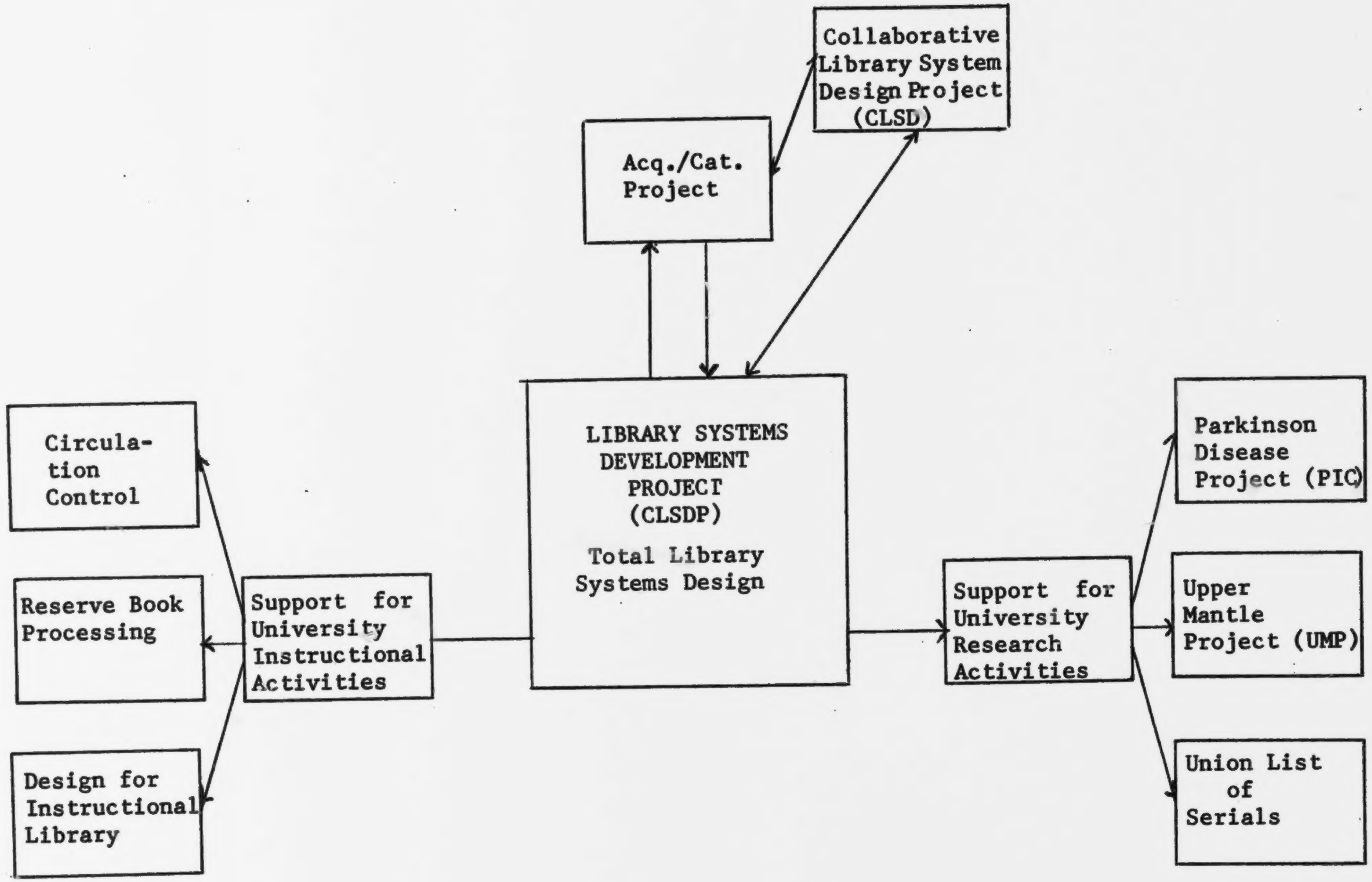
g. 1968-present. Acquisition/Cataloging Project. A systems study of acquisition and cataloging has been done. Emphasis is placed on (1) developing general systems which may have applicability to other institutions, and (2) coordinating development and design work with other large research libraries engaged in similar work. Partially supported by a grant from the National Science Foundation.
Status: Description and analysis of monograph acquisitions procedures completed; preliminary design of a computer based acquisition system completed; description of monograph cataloging procedures initiated; preliminary total systems specifications have been written.

h. 1968-present. Collaborative Library Systems Development Research in the area of computers and generalized library systems undertaken in cooperation with Stanford and Chicago. Objectives: (1) facilitate prompt exchange of working data, (2) explore the feasibility of developing general computer-based systems, and (3) establishing and maintaining liaison with key national agencies. Partially supported by a grant from the National Science Foundation.
Status: Mechanism for collaborative work established; joint specifications for an acquisitions system are being developed.

i. 1967-present. Library Staff Education. Regularly scheduled seminars are given to general library staff on computer technology and systems analysis. Special seminars are given to main library staff for particular computer-based systems.
Status: Periodic seminars held in conjunction with the libraries In Service Training Course; special seminars scheduled and given as necessary.

Columbia University Libraries

Relationship of Automation Projects


Collaborative Library System Design Project (CLSD)
Acq./Cat. Project
LIBRARY SYSTEMS DEVELOPMENT PROJECT (CLSDP)
Total Library Systems Design
Circulation Control
Reserve Book Processing
Design for Instructional Library
Support for University Instructional Activities
Support for University Research Activities
Parkinson Disease Project (PIC)
Upper Mantle Project (UMP)
Union List of Serials

```
       CIRCULATION LIST AS GF    02/12/69      SUPERCEDES    02/12/69          PAGE  0057

        BOOK CALL NUMBER           BORROWER   CHRGD-SEQNCE H  DUE           BOOK CALL NUMBER          BORROWER   CHRGD-SEQNCE H  DUE

QA 0009     .W64 1965              MISSNG     81025-110204    99999 *  U. 0102     .M68                 SKLOOT     90106-110010    99999
QA 0076.5   .J5913 1968            WESTA      80920-120014    99999 *  U. 0103     .M23                 BINDER106  90120-140162    99999
QA 0076.5   .V4                               90205-110169    90310 *  U. 0104     .P35 1968                       90125-120315    90222
QB 0054     .F6 1966               HART       80219-110674    99999 *  U. 0115     .H9 Z73              BINDR      80520-136370    99999
QC 0023     .H74 C2                LILLAN     80926-110104    99999 *  U. 0162     .W9                             90114-120584    90218
QC 0775     .L88 1966 C2           CLAIRETD   80926-120172    99999 *  U. 0162.6   .F55                 BINDR106   81014-140114 W  99999
QD 0024     .B38 N3               BINDER1C6   90113-140151    99999 *  U. 0240     .D2813 1967          MISSNG     81009-110015    99999
QD 0026     .H6 1968              BINDER1C6   81216-140161    99999 *  U. 0768     .M3 1911B            BINDER106  90127-140027    99999
QL 0775     .AR29                             90205-120567    90310 *  U. 0800     .B75                            80313-110333    80417
QL 0775     .L88 1966 C1                      71205-110055    99999 *  U. 0800     .B75                 MISSNG     80802-110032    99999
QP 0026     .S53 A37               BINDR1C6   80527-140050    99999 *  U. 0800     .C57 1967            BASIUK     81219-120749    99999
QP 0352     .N6 1967AA            RABINCVIT   90115-120585    99999 *  UA 0010     .H3                  CATA       80523-110231    99999
R. 0015.73  AM351 V2 1960-64       BINDR      80408-140905    99999 *  UA 0023     .A73                 MISSING    81212-110230    99999
R. 0483     .B6 B8 1943            BINDR106   80304-140440    99999 *  UA 0023     .L48                 HOLD       81115-150062 R  99999
R. 0737     .S48                   RUDY       80809-120147    99999 *  UA 0023     .M25                            90206-120543    90311
R. 747.J62  C42 V2 C2              RCTHAN     80401-110467    99999 *  UA 0646.5   .U5 F6               KRENGEL    90121-120088    99999
RA 0418     .R37 1966             BESEARCH    90128-110084    90227 *  UA 0782     .P3                             90108-120067    90218
RA 0505     .G73                  BINDER1C6   90106-140092    99999 *  UA 0829     .Y8 J8               BINDER106  90203-140477    99999
RC 0351     .B5 1946A              CAMPELL    81127-110190    99999 *  UB 0211     .R92 P47 1968B       WOOD       90109-110072    99999
RC 0438     .R5                   BINDER      90120-134711    99999 *  UB 0270     .D83                 GREEWALD   80416-110272    99999
RC 0455     .R6                   BERNARD     90116-120486 H  90130 *  UB 0270     .D83                 RABIOVIT   80927-120364    99999
RM 0301     .B3                    MISSNG     80613-110009    99999 *  UB 0270     .H6                  RABIOVIT   80927-120360    99999
RM 0301     .B3 C2                            90115-150014    90218 *  UB 0270     .U77 1967                       81205-150027    90116
RR 1905     .W27                              81218-120564    90126 *  UB 0270     .U8                  MISSNG     80924-110007    99999
S. 0469     .H9 S9                 BINDR1C6   81028-140072    99999 *  UB 0270     .W5                             71202-112180    71220
S. 0521     .T1593                BESEARCH    90120-120143    90220 *  UB 0270     .W5 C2               RABIOVIT   80927-120367    99999
S. 0930     .IR5                              90115-150023    90218 *  UB 0271     .R92 P3              WOOD       90109-110071    99999
SB 0063     .L5 D7                BINDER106   81230-140377    99999 *  UB 0271     .R92 S573            BINDER106  81209-140090    99999
SH2         DW5                   HCLD        81126-120282 5  99999 *  UB 0342     .G7 B6               CSHER      90201-120064    90630
SL7         SL163 V3-5 7-10 14 ILLON          80102-110015    99999 *  UB 0343     .M8                  OSHER      90201-120063    90630
T. 0185     .M4                    BINDR1C6   81028-140007    99999 *  UB 0343     .W5                  BINDER106  90120-140173    99999
TF 0145     .K7                   BINDER106   81216-140594    99999 *  UB 0345     .A8 S6               CSHER      90201-120062    90630
TK 0715     .R75 1964              KNOT       80112-110260    99999 *  UG 0446.5   .S6543               BINDER106  90127-140192    99999
TL 0540     .G3 A32                BINDR      71221-110662    80624 *  UH 0224     .UN34 1867-69        LILLAN     80731-110013    99999
TL 0788.4   .K3                    LILLAN     80912-110015    99999 *  V. 0103     .M3                  SLOUA      80701-120106    99999
TP 0548     .S68                  MISSING     80216-110383    99999 *  V. 0210     .S692                BASIK      81010-120155    99999
TRANSFORT TCPICS                   BD         70830-110033    80421 *  V. 0857     .H48                 BASIK      81010-120156    99999
TX 0335     .U55 1967              RESRF      80822-110225    99999 *  VA 0058     .L45 1968B           BINDER106  90120-140148    99999
TX 0637     .B85 V1 1829           SHRDER     80521-110267    99999 *  VA 0429.6   .W4                             90206-150002    90311
TX 0637     .B85 V2 1829           SHRDER     80521-110266    99999 *  VA 0456     .L43                 EVANS      90204-110190    99999
TX 0637     .B85 1967              HEIN       80612-110093    99999 *  VA 0513     .G68 V2              MISSNG     80930-110132    99999
TX 0637     .C5 1967               CARR       80207-111398    99999 *  VA 0926     .R57                            90206-150005    90311
TX 0715     .P468                 WILSON      90111-120071    99999 *  VF 2051     .V3                  HANST      80617-120137    99999
TX 0731     .S53                  GRAFF       90205-120341    99999 *  VK 0016     .A7                             71129-110808    71228
TX 0757     .H68 1968              CATA       81114-120284    99999 *  VK 1061     .G7                  HANST      80617-120141    99999
TX 0819     .A1 O5 1968B           CATA       81114-120285    99999 *  VK 1096     .F7                  HANST      80617-120140    99999
U. 0021.5   .B52                              90124-150021 H  90207 *  VK 1491     .F4                  EVANS      90204-110188    99999
U. 0022.3   .V3                   BINDER1C6   81216-140154    99999 *  VM 0016     .M67                 HENLE      90203-120133    99999
U. 0053     .J6 A3 1967                       90116-150098    90220 *  VM 0980     .C7 H86 1968         BINDR106   81118-140171    99999
U. 0055     .K6 A3 1966           VERWAYEN    81212-110027    99999 *  VX 1407     .B23                 HANST      80617-120139    99999
```

Circulation System:

Sample of Daily Circulation List

```
        PUBLIC RESERVES LIST AS OF  2/06/69

WESTIN, ALAN
    PRIVACY AND FREEDEOM
                   1967
WESTIN, ALAN
    FREEDOM NOW
                   1964
WHARTON, VERNON LANE
    THE NEGRO IN MISSISSIPPI, 1865-1890
                   1965
WHITAKER, ARTHUR AND JORDAN, DAVID
    NATIONALISM IN CONTEMPORARY LATIN AMERICA
                   1966
WHITAKER, ARTHUR P.
    THE UNITED STATES AND ARGENTINA
                   1954
WHITE, HARRISON C.
    THE ANATOMY OF KINSHIP: MATHEMATICAL MODELS FOR STRUCTURES OF
                   1963
WHITE, LEONARD D.
    THE JEFFERSONIANS
                   1956
WHITE, PATRICK C. T.
    A NATION ON TRIAL: AMERICA AND THE WAR OF 1812
                   1965
WHITING, JOHN AND CHILD, IRWING
    CHILD TRAINING AND PERSONALITY-A CROSS CULTURAL STUDY
                   1966
WHITING, JOHN W. M.
    CHILD TRAINING AND PERSONALITY
                   1958
WHITNEY, COURTNEY
    MACARTHUR
                   1956
WHITNEY, SIMON N.
    ANTITRUST POLICIES
                   1958
WHITNEY, SIMON N.
    ANTITRUST POLICIES
                   1958
WHORF, BENJAMIN L.
    LANGUAGE, THOUGHT AND REALITY
                   1959
WHORF, BENJAMIN LEE
    LANGUAGE, THOUGHT, AND REALITY
                   1956
WHYTE, WILLIAM F.
    STREET CORNER SOCIETY
                   1955
WIEBE, ROBERT
    SEARCH FOR ORDER
                   1967
WIEBE, ROBERT
    BUSINESSMEN AND REFORM
                   1962
WIEBE, ROBERT
    THE SEARCH FOR ORDER
                   1967
```

Reserve Book Processing System:

Public Reference List

```
                        COLLEGE LIBRARY                  11/19/68
PROFESSOR: BERMEL          RESERVE LIST FOR: ENG-CL G4221X     FALL 1968

ARDEN, JOHN
    SERJEANT MUSGRAVES DANCE
                1960        B825AR28     W
BENTLEY, ERIC
    LETS GET A DIVORCE AND OTHER PLAYS
                1958        B844.12     B446
BENTLEY, ERIC
    THE MODERN THEATRE
                1954        C808.2     B4464 V6
BENTLEY, ERIC
    THE MODERN THEATRE
                1954        C808.2     B4464 V3
BENTLEY, ERIC
    THE MODERN THEATRE
                1954        C808.2     B4464 V2
BENTLEY, ERIC
    THE MODERN THEATRE
                1954        C808.2     B4464 V1
BENTLEY, ERIC
    IN SEARCH OF THE THEATRE
                1953        C809.2     B443
BENTLEY, ERIC
    THE PLAYWRIGHT AS THINKER
                1955        C809.2     B4411
BENTLEY, ERIC
    BERNARD SHAW
    REV. ED.    1957        C825.SH2     D331
BRUSTEIN, R
    THE THEATRE OF REVOLT
                1964        C809.2     B838
CLURMAN, H
    THE FERVENT YEARS
                1950        B812.128     C628
COLE, TOBY
    PLAYWRIGHTS ON PLAYWRITING
                1960        B808.2     C67631
FERGUSON, FRANCIS
    THE IDEA OF A THEATRE
                1958        C809.2     F38
```

Reserve Book Processing System:

Professors Lists

```
                        DAILY RESERVES WORKLIST OF  1/31/69          SUPPLEMENTS WORKLIST OF  1/22/69
085452     BLONDEL, JEAN                                                                      TERM 691
               VOTERS, PARTIES AND LEADERS                                                       GOVT G6402Y     RASMUSSEN        03
                          1963       C 324.42     B62                                                     SHORTAGE 00  TOTAL NEED 03
                 COPIES  01 02 03 04 05 06 07
                        +  +  +  +  +  +  +
                 TOTAL COPIES 07  ORDR 00

086920     BLONDEL, JEAN
               VOTERS, PARTIES AND LEADERS
                          1963       B 342.42     B62
                 COPIES  01 03
                 TOTAL COPIES 02  ORDR 00

015544     BLOOMFIELD, LEONARD                                                                TERM 683
               LANGUAGE                                                                          ENG G6801X      DOBBIE, E.       06
                          1933       C 401      B623                                                      SHORTAGE 00  TOTAL NEED 06
                 COPIES  01 02 03 04 05 06                                                    TERM 691
                        +  +  +  +  +  +                                                         LING F3101Y     GARCIA           03
                 TOTAL COPIES 06  ORDR 00                                                        LING 4124Y      SILVER           01
                                                                                                          SHORTAGE 00  TOTAL NEED 04

028519     BLUM, JEROME                                                                       TERM 691
               LORD AND PEASANT IN RUSSIA: FROM THE NINTH TO THE NINETEENTH CENTURY              HST F1177Y      RYU              05
                          1964       C 947      B625                                                      SHORTAGE 02  TOTAL NEED 05
                 COPIES  01 02 03
                        +  +  +
                 TOTAL COPIES 03  ORDR 00

028542     BLUM, JOHN MORTON                                                                  TERM 691
               THE REPUBLICAN ROOSEVELT                                                          HST F1110Y      GARRATY          10
                          1954       C 92R67     B61                                             HST G8680Y      GARRATY          03
                 COPIES  01 02 03 04                                                                      SHORTAGE 09  TOTAL NEED 13
                        +  +  +  +
                 TOTAL COPIES 04  ORDR 00

086659     BLUM, JOHN MORTON                                                                  TERM 691
               WOODROW WILSON AND THE POLITICS OF MORALITY                                       HST C3130Y      KERBY            03
                          1956       C 92W69     B4                                                       SHORTAGE 00  TOTAL NEED 03
                 COPIES  01 02 03 04 05 06 07
                        +  +  +  +  +  +  +
                 TOTAL COPIES 07  ORDR 00

086660     BLUM, JOHN MORTON                                                                  TERM 691
               WOODROW WILSON AND THE POLITICS OF MORALITY                                       HST C3956Y      MINER            02
                          1956       B 973.82       W6955                                                 SHORTAGE 02  TOTAL NEED 02
                 TOTAL COPIES 00  ORDR 00

025652     BLUNT, ANTHONY                                                                     TERM 691
               ARTISTIC THEORY IN ITAL 1450-1600                                                 FA F1001Y       BECK-BLETTER     10
                          1964       C 701      B62811                                           FA F1001Y       LUNDE            09
                 COPIES  01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17                               SHORTAGE 02  TOTAL NEED 19
                        +  +  +  +  +  +  +  +  +  +  +  +  +  +  +  +  +
                 TOTAL COPIES 17  ORDR 00

084593     BOBER, MANDELL, MORTON                                                             TERM 691
               INTERMEDIATE PRICE AND INCOME THEORY                                              ECON G4228Y     HEILBRUN         04
                          1962       C 338.5     B63                                                      SHORTAGE 04  TOTAL NEED 04
                 TOTAL COPIES 00  ORDR 00
```

Reserve Book Processing System:

Librarians's Work List

```
     WEEKLY RESERVES WORKLIST OF  8/15/68        REPLACES ALL PREVIOUS WORKLISTS
                                                                                  SHORTAGE 00  TOTAL NEED 01
                1967        D338.5   ST512                  BURGESS
           COPIES  01
                   +
           TOTAL COPIES 01  ORDR 00
                                                                           TERM 681
001256    STIGLER, GEO.                                                      EC F3211Y       SHAMI            02
             THE THEORY OF PRICE                                                      SHORTAGE 00  TOTAL NEED 02
                1967        D338.5   ST512
           COPIES  01 02
                   +  +
           TOTAL COPIES 02  ORDR 00
                                                                           TERM 681
000800    STIGLER, GEORGE                                                    EC C3070Y - SEC FRIEDMAN         06
             THE THEORY OF PRICE                                                      SHORTAGE 06  TOTAL NEED 06
             REVISED OR             D338.5   ST511
           TOTAL COPIES 00  ORDR 00
                                                                           TERM 681
000522    SUITS, DANIEL B.                                                   EC-CC F1411Y    BURTLE           05
             STATISTICS   AN INTRODUCTION TO QUANTITATIVE ECONOMIC RESEARCH           SHORTAGE 01  TOTAL NEED 05
                1963        D311     SU39                   BURGESS
           COPIES  01 02 03 04
                   +  +  +  +
           TOTAL COPIES 04  ORDR 00
                                                                           TERM 681
000355    TINBERGEN, J.                                                      EC-CC 3021Y     LUBITZ           02
             DESIGN OF DEVELOPMENT                                                    SHORTAGE 00  TOTAL NEED 02
                1958        D338     T49
           COPIES  01 02 03 04 05 06 07
                   +  +  +  +  +  +  +
           TOTAL COPIES 07  ORDR 00
                                                                           TERM 681
002500    TRIFFIN, ROBERT                                                    EC C3041Y                        07
             GOLD AND THE DOLLAR CRSIS                                                SHORTAGE 00  TOTAL NEED 07
                            D332.4   T731
           COPIES  01 02 03 04 05 06 07
                   +  +  +  +  +  +  +
           TOTAL COPIES 07  ORDR 00
                                                                           TERM 681
000622    U.S. CONGRESS.  JOINT ECONOMIC COMM. SUBCOMM. ON ECONOMY IN GOVT.  EC C3062Y       HARRISS          06
             REPORT ON ECONOMY IN GOVERNMENT                                          SHORTAGE 00  TOTAL NEED 06
                JULY, 1967
           COPIES  01 02 03 04 05 06
                   +  +  +  +  +  +
           TOTAL COPIES 06  ORDR 00
                                                                           TERM 681
000633    U.S. CONGRESS. JT. ECON. COMM. SUBCOMM. ON ECONOMY IN GOVT.        EC C3062Y       HARRISS          01
             BACKGROUND MATERIAL ON ECONOMY IN GOVERNMENT                              SHORTAGE 00  TOTAL NEED 01
                6967
           COPIES  01
                   +
           TOTAL COPIES 01  ORDR 00
                                                                           TERM 681
001212    VICKREY, WILLIAM                                                   EC F3211Y       SHAMI            05
             MICROSTATICS                                                             SHORTAGE 00  TOTAL NEED 05
                1964        D330.1   V66
           COPIES  01 02 03 04 05
                   +  +  +  +  +
```

1967 DOCUMENT CITATIONS

03201pc967
Farrer, P.
Mistilis, S.P.
Absorption of exogenous and endogenous biliary copper in the rat
Nature 213(5073):291-292, Jan. 1967.
6 Refs. /Experimental/ Eng. Gt. Britain
Copper - metabolism
Intestinal absorption
Copper - classification
Copper - measurement

03202pc967
Bolt, W.L. De
Movement epilepsy: two case reports with photographs of the typical movements.
Bulletin of the Los Angeles Neurological Societies 32(1):1-5, Jan. 1967.
9 Refs. /Case Report/ Eng. U.S.A.
Movement
(Seizures) - etiology
Athetosis

03203pc967
Andrews, J.M.
Neurological disease on Guam: a review of past and present investigations.
Bulletin of the Los Angeles Neurological Societies 32(1):30-42, Jan. 1967.
60 Refs. /Review/ Eng. U.S.A.
Nervous system diseases - history
Amyotrophic lateral sclerosis - history
(ALS/P-D) - history
Amyotrophic lateral sclerosis - geographic distribution
Epidemiology

03204pc967
Feinstein, B.
Levin, G.
Alberts, W.W.
Wright, E.W.,Jr.
Stereotaxic therapy for dyskinesias and
tion of clinical results.
Bulletin of the Los Angeles Neurological Societies 32(1):55, Jan. 1967. Abstr.o
0 Refs. /Clin. Study/ Eng. U.
(Dyskinesia) - surgery
Radio waves

03205pc967
Markham, C.H.
Clinical pathological correlations of stereotaxic lesions in Parkinson's disease and other movement disorders.
Bulletin of the Los Angeles Neurological Societies 32(1):55-56, Jan. 1967. Abstr.only.
0 Refs. /Clin. Study/ Eng. U.S.A.
(Parkinson's disease) - surgery
(Dyskinesia) - surgery
(Nucleus ventralis thalami lateralis) - lesions
(Nucleus subthalamicus) - lesions
(Putamen) - lesions
(Parkinson's disease) - physiopathology

03225pc967
Ojemann, G.A.
Buren, J.M. Van
Respiratory, heart rate, and GSR responses from human diencephalon.
Archives of Neurology 16(1):74-88, Jan. 1967.

cont. next column

31 Refs. /Clin. Study/ Eng. U.S.A.
Respiration
Galvanic skin response
Heart rate
Diencephalon - electric stimulation

1966 DOCUMENT CITATIONS

The following citations represent documents published in 1966 which have been recently received or identified by the Center and are listed here for the first time.

03149pc966
Herz, A.
Zieglgaensberger, W.
Synaptic excitation in the corpus striatum inhibited by microelectrophoretically administered dopamine.
Experientia 22(12):839-840, Dec. 1966.
12 Refs. /Experimental/ Eng. Switzerland
(Synaptic activity)
Dopamine - pharmacodynamics
(Inhibition)
(Nuclei intralaminares thalami) - electric stimulation
Amino acids - pharmacodynamics

03150pc966
Anden, N.-E.
Fuxe, K.
Larsson, K.
Effect of large mesencephalic-diencephalic lesions on the noradrenalin, dopamine and 5-hydroxytryptamine neurons of the central nervous system.
Experientia 22(12):842-843, Dec. 1966.
11 Refs. /Experimental/ Eng. Switzerland
Mesencephalon - lesions
Diencephalon - lesions

Parkinson Information Center

Sample Page from The Bi-Weekly

Announcement Bulletin

n's

9 Refs. /Experimental/ Eng. Gt. Britain
Hepatolenticular degeneration - drug therapy
Chelating agents - therapeutic use
Chelating agents - administration & dosage

03172pc966
Koff, G.Y.
Langfitt, T.W.
Tremorine-induced rage and the limbic system.
Archives Internationales de Pharmacodynamie 164(2):272-285, Dec. 1966.
38 Refs. /Experimental/ Eng. Belgium
Tremorine - pharmacodynamics
Behavior,animal - drug effects
(Lesions,experimental)
(Psychological reactions)
(Oxotremorine) - pharmacodynamics
Limbic system - lesions

REFERENCES

Polish progress report 1966-1967, for the Upper Mantle Project. Warsaw, Aug. 1967.
9p.
typescript.
At head of title: Polish Academy of Sciences. National Committee for Geophysics and Geodesy. Polish Upper Mantle Commission.
Presented to the International Committee on the Upper Mantle Project for the XIV General Assembly of the International Union of Geodesy and Geophysics.
UMP Polish progress report.

GENERAL: SOUTH AFRICA

AV4-01190
International Upper Mantle Project. South African Upper Mantle Committee.
Upper Mantle Project. Report by the South African National Committee, August 1967.
2 l.
typescript.
UMP South African progress report.

GENERAL: SPAIN

AV6-01191
International Upper Mantle Project. Spanish Upper Mantle Committee.
Preliminary report of the seismological activities related with the UMP. Madrid, Jul. 1967.
4 l.
At head of title: ICSU, Grupo de Trabajo en el Proyecto del Manto Superior.
typescript.
UMP Spanish progress report.

GENERAL: SWITZERLAND

AW5-01206
Thams, J. C. ed.
Le developpment de la geodesie et de la geophysique en Suisse. The development of geodesy and geophysics in Switzerland. Zurich, 1967.
98p. illus., maps (some fold.). ENG or FRE
Commemorative book presented to the participants in the XIV General Assembly of the International Union of Geodesy and Geophysics by the Swiss Academy of Natural Sciences.

GENERAL: TURKEY

AX6-01192
International Upper Mantle Project. Turkish Upper Mantle Committee.
Activities connected with Upper Mantle Project in Turkey. Progress report: September 1967. Sept. 1967.
3 l.
typescript.
UMP Turkish progress report.

GENERAL: U.S.A.

AY5-01013
Neal, J.T. ed.
Playa surface morphology: miscellaneous investigations. Editor: James T. Neal, Capt., USAF. Contributors:D. Carpenter, R.Z. Gore, D.B. Krinsley, W.S. Motts, J.T. Neal, G.E. Stoertz [and] C.C. Woo. Mar. 1968.
151p. illus. maps. (1 fold.).
U.S. Air Force Cambridge Research Laboratories. Environmental Research Papers, no.283.
AFCRL-68-0133.
Carpenter, D. Gore, R.Z. Krinsley, D.B. Motts, W.S. Stoertz, G.E. Woo, C.C. U.S. Air Force. Cambridge Research Laboratories. Terrestrial Science Laboratory.

AY5-01088
Hamilton, W.
Geologic and crustal cross section of the United States along the 37th parallel. A contribution to the Upper Mantle Project. Washington, D.C., U.S. Geological Survey, 1965.
col. map. 82 x 98 cm.
Miscellaneous Geologic investigations map I-448.
Scale 1:2,500,000.
Pakiser, L.C.

AY5-01094
Knopoff, L.
Upper Mantle Project: Phase III, 1968-1970. American Geophysical Union, Transactions 48(2):757-8, June 1967.

GENERAL: U.S.S.R.

AY8-01193
International Upper Mantle Project. U.S.S.R., Upper Mantle Committee.
Provisional plan for works in the USSR for the period 1968-1970. Moscow, 1967.
5 l.
On cover: USSR National Committee on the Upper Mantle Project.
typescript.
UMP U.S.S.R. progress report.

AY8-01197
Akademiia Nauk SSSR
Kory i verkhniaia mantiia zemli; bibliograficheskii ukazatel' 1960-1964. [The crust and upper mantle of the Earth; bibliography 1960-1964 Moskow, Nauka, 1967.
175p. RUS
At head of title: Sektor Seti Spetsial'nykh Bibliotek . Biblioteka Instituta Fiziki Zemli im. O.Iu. Shmidta.

PETROLOGY AND MINERALOGY: CHILE

BD9-01172
Katsui, Y.
Geology of the neo-volcanic area of the Nevados de Payachata; (Provincia de
[1967]
4 l. illus., col. fold. m
Chilean Committee of the
Progress report, no.1, Geol
typescript.
Gonzalez, O.

Upper Mantle Project:

Computer Formatted Book Form

Catalog Page

PETROLOGY AND MINERALOGY: FRANCE

BG9-01104
Anthonioz, P.M.
Geologie sommaire de la region de Morais (Tras-os-Montes, Portugal). [Geological summary of Morais area (Tras-os-Montes, Portugal)] Leidse Geologische Mededelingen, 36:301-4, Dec.15, 1966.
FRE
Presented at : "Primera reunion sobre geologia de Galicia y norte de Portugal", Santiago de Compostela (Prov. La Coruna), September 6-14, 1965.
English abstract.

PETROLOGY AND MINERALOGY: NETHERLANDS

BQ4-01043
Roever, W.P. de
Preliminary note on ferrocarpholite from a glaucophane-and lawsonite-bearing part of Calabria, Southern Italy. Koninkl. Nederl. Akademie van Wetenshappen, Proceedings, series B, 70(5):534-7, 1967.
Roever, E.W.F. de Beunk, F.F. Lahaye, P.H.J.

BQ4-01044
Roever, W.P. de
Overpressure of tectonic origin or deep metamorphism? [n.p., 1967]
7 l.
typescript.
Translation of "Overdruk van tektonische oorsprong of diepe metamorfose?." Koninkl. Nederl. Akad. Wetensch., Versl. Grew. Vergad.

Stanford University Libraries
Project BALLOTS
A Summary

Allen B. Veaner
Assistant Director for Automation
Stanford University Libraries

A Paper Prepared for the Stanford Conference on Collaborative Library System Development, October 4-5, 1968

In this summary, I propose to cover the following: a brief outline of how our project is organized, the fundamental assumptions behind the desire to establish an on-line bibliographic control system, a summary of what has been accomplished to date, some conclusions, and a few highly speculative remarks about the future.

The rise of computer science and information science is demanding a response from traditional library thinking. How should libraries respond to the powerful innovative forces now at work?

Stanford would like to be in the forefront of these innovative developments, and has chosen to evolve its information system design by merging two large scale bibliographic retrieval projects. The first of these is SPIRES -- originally an acronym for the Stanford Physics Information Retrieval System -- now enlarged in scope as the Stanford Public Information Retrieval System. The second is BALLOTS, standing for Bibliographic Automation of Large Libraries Using Time-Sharing. Begun in February, 1967, with funding from the National Science Foundation, SPIRES aimed at providing on-line searching of a data base at first consisting of citations describing a collection of preprints in high energy physics. Professor Edwin B. Parker, of Stanford's Institute for Communication Research, is Principal Investigator for SPIRES.

Just as SPIRES was being funded, the Library was independently seeking aid from the Office of Education to establish a bibliographic control system to be implemented in two phases. The first objective was to establish a computerized, internal technical processing system for the Library's traditional functions, such as acquisition, cataloging, circulation, and serials control; the second objective was to extend bibliographic services of greatly enlarged scope to the academic public. It was perhaps inevitable that these two projects should join forces, and to assure that, the Library asked Ed Parker to join the project's Faculty Advisory Committee. The Library maintains a task force responsible for analysis of existing operations and design of new systems, while the task force associated with the Institute for Communication Research is responsible for systems software and applications programming. The joint projects are overseen by an Executive Committee chaired by Professor William Miller, Associate Provost for Computing; members are Ed Parker, Ed Feigenbaum, Director of the S.C.C.; Rudy Rogers, and myself. We believe that it makes a great deal of sense for librarians, behavioral scientists, and information scientists to work together.

The fundamental problem in bibliographic access is the communication of bibliographic messages to a user. Among the issues surrounding this problem are: the nature of the dictionary catalog, national standardization, decentralization of bibliographic access, general applicability of system design, regionally shared data bases, servicing of multiple data bases, and economics.

The characteristics of large card catalogs in dictionary form are well known: an alphabetico-logical organization which denies the "dictionary" appellation, which chains the user to a mysterious and ill communicated filing algorithm, and which provides him very little flexibility in formulating searches. In short, the card catalog provides only undirectional communication. We would like to establish a two-way communication system so that the user can conduct his searches interactively. We propose to provide the searcher with an interactive visual terminal, rather than a typewriter terminal which is too slow an output device for bibliographic messages.

We further propose to accept Library of Congress cataloging as a true national standard, suggesting that if there is a pressure for change, it should be in the direction of conforming local practice to the national standard, and not the reverse. We are not unaware of the attendant operational and political difficulties in actually accomplishing this.

We would like to establish a publicly accessible, computer-maintained, central bibliographic file. Internally, this can free our technical processing staff from the constraints of a single location manual file and allow greater flexibility for locating staff. Externally, we would like to service any Stanford library and any user having access to a terminal. We have already begun to discuss with the Law Library the possibility of integrating their acquisition work with the Main Library's.

In the development phase, there will be no computer in the Library. The Library and the Institute for Communication Research are working closely with the S.C.C. in the belief that the first task is to find the right technical solution. The hardware utilized is the Campus Facility's IBM 360/67. Whether a separate, dedicated machine is the best long-run solution is a question that must be deferred until the correct technical solution is identified.

One final issue - dare I mention it? - lurks very visibly in the foreground, and that is economics. We grossly underestimated the cost of machine time for a development project. Fortunately, we were able to renegotiate our budget and shift salary savings into computer services. Now half of our budget is in that category. However, such a shift can be meaningful only in a context where there exist superbly qualified systems programmers, working in an outstanding intellectual environment. The range and complexity of services offered by the S.C.C. is rivalled by few other organizations, and we are pleased to be associated with this imaginative group.

I would now like to describe briefly our project's activity during the past 15 months.

We have assembled a stimulating combination of librarians, system analysts, and systems programmers. A good deal of healthy interaction has ensued - on one side, we have conveyed an appreciation of the complexity of bibliography; on the other side, we have learned to give up our "catalog card mentality", preoccupation with filing rules and other inhibiting or retrograde influences.

We next carried out a detailed systems analysis of our present procedures and learned the usual, startling facts that normally have low visibility: outmoded or unnecessary procedures, files that were maintained for no purpose, and so forth. Appropriate changes were made and some minor immediate benefits achieved. Next, working with the first line of users, our own librarians, we worked out a set of system requirements -- tasks that a future system, whether automated or not, needed to fulfill. Finally, a design was evolved around these requirements. To reach this point required an investment of about ten man-years, including the contributed time of the regular library staff.

The design is based upon a series of time-dependent events which correspond roughly to the traditional functions: acquisition, cataloging, and circulation. (Design effort for serials control will be deferred until all other systems are operational for several reasons: 1. We believe that control of serials represents the most difficult and challenging facet of library automation. 2. We want to take advantage of the work of the National Serials Data Program. 3. We wanted some prior working experience before plunging into serials.)

The heart of the design is the MARC record to be provided by the Library of Congress. Using MARC, we propose to pre-catalog incoming materials wherever possible, keyboarding only those entries for which MARC data is not found within some reasonable time. We propose to maintain three machine-readable bibliographic files and have storage capacity enough for an estimated two years' cumulation: the files are the MARC data, an In Process File, and the start of a machine-readable catalog or holdings file. From these files we are preparing to support the following services: file building (with edit checks), on-line, interactive searching from visual or typewriter terminals, file updating, and a variety of printed outputs: lists, catalog cards, purchase orders, and management reports.

The staff of SPIRES has developed a relatively natural command language. The SEARCH command specifies the data base to be serviced; the FIND command specifies the appropriate index file. Data found in a MARC file can be copied into the In Process File by entering the command COPY followed by a record identification number, i.e., the Library of Congress card number. The usual logical operators (AND, OR, NOT) are available, as are arithmetic comparisons for date searches.

The first draft of a User's Manual has been developed, and a start has been made at setting up a consulting service to aid library and other prospective users. Within the acquisition function, specific written procedures are now being worked out to guide staff members who will operate the automated acqusition system. A Data Control Function has been defined and established to oversee all input and output, control forms, handle distribution and mailing, as well as assist in training terminal operators.

Finally, with the aid of the Stanford Computation Center, we are on the point of selecting a visual terminal which we hope may become an interim campus standard. One particularly attractive terminal has the facility to display not only text, but also graphics and pictorial data, such as TV, facsimile, etc. Our commitment to visual displays is sufficiently strong that cable is now being pulled to connect the Computation Center and the Library; we expect this work to be completed around November 15.

We have concluded that pioneering an on-line bibliographical control system in a large research library is difficult and expensive. At worst, however, it sometimes appears that in maintaining our manual systems, we are already paying the cost of library automation without achieving any of its benefits. Librarians are sometimes urged to wait and see, because we are told each year that the cost of computation is coming down. That's true: the unit cost of a cycle of computer time is coming down, but so is the unit cost of photocopying and telephone communication -- yet our total budgets in those categories continue to rise, simply becuase we keep spending more just because these services have become so inexpensive. So, at the present time, it is apparent that in a development project, the dollars for machine time compete on more than equal terms for personnel dollars. I have already mentioned that half of our budget is allocated to machine time.

We also conclude that a highly generalized, flexible record design is most advantageous in a development project and well worth the extra overhead cost. To be free of fixed length limitations simplifies design change, and any development project must be prepared for frequent changes.

The stimulus of the non-librarian has been of immense significance in this project -- particularly that derived from our own staff of analysts and the systems programmers at the Computation Center. However, it is absolutely essential for the librarian to learn the new technology for himself. He cannot abrogate this responsibility. Incidentally, in working with a new technology, it is well to remember that the librarian may have as much -- if not more -- to unlearn as he has to learn. We simply must free ouselves from the fetters of traditional concepts of file organization and filing rules.

Perhaps our most exciting and refreshing conclusion is that we know multiple-terminal, on-line searching works -- we've demonstrated it -- and that it's going to represent a really significant breakthrough -- for technical processing today, and for the user tomorrow. None of this will be meaningful, however, unless the Library of Congress can deliver the goods -- in the form of rapidly disseminated, standardized bibliographic data. Nothing must interfere with that mission, and not just for Stanford's sake.

A few words about our future activities. We propose to continue development in the sequence already established, and go on to cataloging, circulation, and serials. Meantime, we will be looking into the mass storage problem for static bibliographic data. The answer may lie in some form of computer-controlled microstorage or in a photodigital store. We would also like to think about the text access problem and will certainly be watching Project INTREX's experience.

In James Dolby's final report to the Office of Education, An Evaluation of the Cost and Utility of Computerized Library Catalogs, the author emphasizes as his primary conclusion "that mechanization of the cataloging function is not only necessary and desirable, but also inevitable." In the Museum of History and Technology at the Smithsonian Institution, a visitor can see many ancient relics of pre-computer civilizations, including such representatives of the paleo-computer era as Howard Aiken's MARK I, the ENIAC, SEAC, and UNIAC. Some of the equipment is less than ten years old. I should like to pose the question whether ten or twenty years hence the Smithsonian might justly display artifacts representing today's bibliographic apparatus in the research library. One widely quoted librarian is alleged to have said, "When the feeling to automate overcomes you, lie down until it goes away." Automation and computers are not going to go away, and we at Stanford had better not lie down.

Collaborative Library Systems Development (CLSD)

Dr. Richard H. Logsdon
Director
Columbia University Libraries

A Paper Prepared for the Stanford Conference on Collaborative Library Systems Development, October 4-5, 1968

Rogers: We're only two minutes late. Our next speaker is a product of Upper Sandusky, Ohio. Dr. Logsdon received his A.B. and B. S. in Library Science from Western Reserve University. He was granted the Ph.D. from Chicago in 1942. He was a college librarian in Colorado and Virginia from 1934 to 1943. From 1942 to 1945 he was professor and head of the Department of Library Science at the University of Kentucky. Part of this time he was on leave with the U.S. Navy, as I well know. In 1947 he became Assistant Director of the Library Science Division of the Veterans Administration, but soon left to become Assistant Director of the Columbia University Libraries, a position he held for five years before becoming Director in 1953. Dr. Logsdon has served as Chairman of the University section of the Association of College and Research Libraries, and as Chairman of the Association of Research Libraries. He's also been active in various boards and committees concerned with education for librarianship, and with Slavic and East European library resources. From 1960 to 1963 he was Chairman of the Library Advisory Committee, Council of Higher Educational Institutions in New York City. He has participated in surveys in many college and university libraries in the Middle West and East, as well as Canada and Puerto Rico. He's presently a member of the Regents Advisory Council in New York State and of the Commissioner's Committee on Library Development. Dick, come tell us about the Collaborative Library Systems Development Project.

* * * * * * * * * * *

Thanks, Rudy. I, too, would like to start with a bit of history, as far as CLSD is concerned, and hope it will not be quite like that course on the French Revolution that Pierce Butler used to tell us about at Chicago. The professor opened the course by saying "before we get into the meat of the main subject, let me fill you in a bit on the background that led to this revolution."

Some seventeen weeks later he would finish the course with, "Now that I have filled you in on the background, I trust that you will do well on the examination next week, which as you know will be on the effects of the French Revolution on western society."

I do not have access to all of the documents because Columbia came rather late in the sequence of events which led to this grant and to the formation of what we call the Collaborative Library Systems Development Project or CLSD for short. From conversations and some documentation, I have concluded that the concept of CLSD developed during 1966 when a number of individuals in and out of the government became increasingly concerned about undue duplication of effort in the necessary but expensive research and development work associated with automation of library activities. Something like CLSD was viewed as a mechanism for testing and demonstrating the advantages of cooperation in these efforts.

There was equal concern of course, that systems developed independently under different grants would (a) be reasonably compatible; and (b) have applications beyond the particular institution. Then (as now) there was the hope that systems of general applicability might be possible.

There were probably other reasons, including the need to set a limit on the number of grants given in sequence anticipating that surely the "nth" NSF grant would be more duplicative of effort than the "nth" minus one or two. Concurrently was the concern that acceptance of a government grant would carry with it the public responsibility to share--through hospitality to visitors, correspondence, and other forms of communication to the profession--interim plans, developments and findings to the point that a grant could become a liability. CLSD was viewed (a) as a means of institutionalizing collaborative efforts among the three participants; and (b) as a formalized procedure for maintaining liaison with other research efforts and the profession generally.

In any event, the concept of a joint and then later collaborative effort became visible in early 1967 with informal queries to a number of potential participants. Somewhat later, in 1967 the fact of Chicago's having its National Science Foundation grant; Stanford its HEW grant; and Columbia an HEW grant for the reserve book system, Columbia became a potential third party to the collaboration. A National Science

Foundation grant to Columbia for its own research and development program[1] followed shortly together with the separate grant from the same agency for CLSD. Chicago was the intended base; it came to Columbia by mutual agreement, primarily because of the availability of office space there. Confirmation of the NSF-CLSD grant was not received until March 1968, some six months ago. However, three-way discussions began earlier with respect to the methods of collaborating on the assumption that the grant could come through.

The objectives of CLSD as stated in the grant request[2] are: (a) "The prompt exchange of working data, information, and ideas among the participating institutions, (b) Providing the means for exploring and arriving at general agreements, where appropriate and possible, on coordination of schedules, and cooperation in approach on specific common objectives, and (c) Providing a better means than now exists for liaison with key national agencies (and of course by inference with the profession at large.) The grant is modest in amount - $60,700. It provides for a Planning Council consisting of the library directors of the three institutions (Fussler, Logsdon and Rogers) and the three technical directors (Fasana, Payne and Veaner). It provides also for a modest amount of travel for occasional meetings of the Planning Council and for more frequent meetings of technical personnel. Meetings are rotated among the three institutions as a means of periodically involving members of the local staffs. Liberal sharing of working documents developed in the several projects supplement the exchange of information at meetings. In addition, it is planned that conferences of this kind and publication of proceedings will serve to share findings with the profession at large. The grant provides for an executive secretary to the Planning Council on a part-time basis. Paul Fasana serves in this capacity. Accomplishments to date, in addition to the substantial interchange of information between and among the three participating institutions are recorded in progress reports to the National Science Foundation.[3]

---

[1] Columbia University. The Libraries. Library system development for a large research library; a proposal for research and/or related activities submitted to the National Science Foundation. January 1, 1968.

[2] Columbia University. The Libraries. Collaborative program in library system development; a proposal for research and/or related activities submitted to the National Science Foundation. February 1, 1968.

While it would be premature to predict developments for the future we believe that CLSD does provide an effective mechanism for sharing experience internally and that we will more than meet our obligation to share findings through conferences of this kind, official reports, informal consultations and correspondence.

---

[3] Columbia University. The Libraries. Collaborative program in library system development; progress reports 1-2 for the period 1 February 1968 to 1 August 1968.

National Collaboration and the National Libraries Task Force: A Course Toward Compatibility

Samuel Lazerow
Chairman, National Libraries
Task Force on Automation
Library of Congress

A Paper Prepared for the Stanford Conference on Collaborative Library Systems Development, October 4-5, 1968

Rogers: Our next speaker is Chairman of the U.S. National Libraries Task Force for Automation and Other Cooperative Services. He has the unusual distinction of having worked in the three national libraries of the United States. From 1947 to 1952 he was Chief of Acquisitions of the National Agricultural Library. From 1952 to 1965 he served successively as Chief of Acquisitions and Chief of the Technical Services Division of the National Library of Medicine, where he participated in the development of a mechanized system for the library's technical operations. He joined the staff of the Library of Congress in 1965. After a survey of the work of the Serial Record Division in 1966 he was named Chief of that Division. He's a graduate of Johns Hopkins and Columbia, and has done advanced study in public administration and technical management.

* * * * * * * * * * *

Thank you Rudy. I think you can set your watch by the way Rudy Rogers runs the meeting. I think I will, too. I think that if I don't get out of here by twelve o'clock, he'll drag me out, I'm sure. I want to add one note to the business of the TV monitors and the World Series and the football games and so on. I think I'd like to add one other facility to this TV business, and that is a closed circuit situation where people like me and Dr. Adkinson, who are cigar addicts, can sit back and talk to you and be talked at, because I don't know how he's faring, but I'm beginning to exhibit withdrawal symptoms. If you see some peculiar gyrations, that's what that means, but regardless of this I am delighted to have this opportunity to meet with you today to learn more about the far-reaching plan Columbia, Stanford, and the University of Chicago Libraries have embarked upon and to acquaint you with some of the current labors of the U.S. National Libraries Task Force as it pursues its basic objective of extending and strengthening the collective system of the three national libraries of the United States.

Cooperation among the national libraries is not a new or recent concept although it has never been pitched at as high a level as at present. Earlier examples include the following: As early as 1901, the Librarian of Congress, in reporting to the Cogress on the state of the Library's collections, commented that few books had been purchased in recent years for Agriculture "because the well organized library of the Department of Agriculture is adequate to the demands," and, with reference to materials in Medicine and Surgery, he explained that "Owing to the accessibility of the library of the Surgeon General's Office and its liberal administration, there has been little expenditure by the Library of Congress in thse lines."[1]

In 1944 the Army Medical Library (now the National Library of Medicine) joined with the Library of Congress in a "systematic review of the classification schedule for medicine."[2]

Since 1945 the Library of Congress has recognized NAL's responsibility to collect comprehensively in agriculture and its allied fields and NLM's similar responsibility for broad coverage in medicine and its allied fields.[3]

The largest single contributor of cooperative cataloging copy to the Library of Congress in 1948 was the Army Medical Library "in accordance with an agreement reached the previous year, according to which this library took principal responsibility for the cataloging of medical books..."[4]

And so it has gone, as the three national institutions have endeavored to advance their services by combining and sharing resources and skills whenever possible and appropriate.

---

[1] Report of the Librarian of Congress, 1901, Washington, D.C., p. 319-320.

[2] Report of the Librarian of Congress, 1944, Washington, D.C., p. 79.

[3] Letters from Librarian of Congress to Army Medical Library and Department of Agriculture Library, February 23, 1945 and October 24, 1945 respectively.

[4] Report of the Librarian of Congress, 1948, Washington, D.C., pp. 92 - 93.

2

As we all know, the reasons behand these collaborative efforts are even more compelling today--the great quantities of material being generated in every field of knowledge; the accelerating costs of acquiring, accessioning, cataloging, and servicing these expanding collections; the mounting pressure from scientists, other scholars and users to have quick access to information; the increasingly difficult task of providing interdisciplinary linkages.

It is the increasing urgency of these problems that has led today's librarians to recognize that some traditional library methods are inadequate and that they must look to the new technology for some positive remedies.

In June 1967 the directors of the three national libraries announced in San Francisco during ALA's annual conference, their institution of a coordinated national library effort "to speed the flow of research information to the Nation's libraries and to the scholars and researchers who use them.[5]

At a press conference at that time, these directors announced their agreement on adoption of "common goals as each proceeds to automate."

They pointed out on that occasion that "this effort to achieve systems compatibility at the national level has far-reaching implications for library automation and library systems of the future."[6]

The broad purpose of the program, as defined by the directors, is to improve access to the world's literature in all areas of human concern and scholarship, so that comprehensive access to the materials of learning can be afforded to all citizens of the United States."

Specific goals indicated in the joint announcement were "the development of a national data bank of machine-readable cataloging information" and a "national data bank of machine-readable information relating to the location of hundreds of thousands of serial titles held by American research libraries," along with the essential objective of achieving compatibility in as many areas of the three libraries" operations as possible.

---

[5] Library of Congress Press Release 67-33, Washington, D.C., June 26, 1967.

[6] Ibid.

Our Task Force was announced at that time as the vehicle for guiding this cooperative effort.

The Task Force (composed of one member and one alternate from each of the three national libraries)[7] has identified specific problem areas requiring detailed study and has named working groups to go into these problems in depth.

Currently ten working groups are active in the following areas:

1. Acquisitions
2. Bibliographic Codes
3. Character Sets
4. Descriptive Cataloging
5. Generalized Output
6. Machine-Readable Format
7. Name Entry and Authority File
8. Serials
9. Subject Headings
10. Systems

All groups have made important progress, as will be evident from the accomplishments to be outlined here.

Each group is chaired by a national library staff member knowledgeable in the problem area concerned, and the memberships are composed of staff having responsibilities in the pertinent areas in their respective national libraries.

Determination of mission statements for each group was a first order of business.

Meetings are held weekly or at the call of the group chairmen who report frequently to the Task Force in brief written reports or in oral presentations.

Last June an all-day session with all group chairmen, at which we were privileged to have Mr. Fasana present, brought the Task Force up-to-date on the progress of each group and provided an opportunity for a profitable exchange among the groups themselves.

---

[7] Task Force members, in addition to Mr. Lazerow, are Bella E. Shachtman, National Agricultural Library, and Samuel Waters, National Library of Medicine, who has just succeeded James P. Riley, Alternates are Mrs. Henriette D. Avram, Library of Congress, Abraham Lebowitz, National Agricultural Library, Stanley Smith, National Library of Medicine. Mr. Irvin J. Weiss, Library of Congress, assists the Task Force. Mrs. Marlene D. Morrisey, Executive Assistant to the Librarian of Congress, is serving as staff assistant to the chairman.

You can well understand that a number of the difficult problems cut across several areas and it is important for groups to be aware of developments in areas other than those of immediate concern.

The automation of serial controls, for example, while the major concern of the Serials Group, involves the groups on Character Sets, Generalized Output, and Machine-Readable Format as well.

We have not yet worked out an entirely satisfactory mechanism for assuring referral of related problems from group to group, but we have found frequent joint discussion and reporting is one useful approach.

The Task Force itself meets weekly for two or more hours of discussion on a variety of topics ranging from compatibility in filing rules to procedures and steps leading toward conceptualization of a hypothetical working system.

An Advisory Committee, composed of representatives from major professional societies, has met once with the Task Force and once in executive session. Jim Skipper is Chairman of the Committee. Its primary purpose is to assist in communications to and from the library community and to give the Task Force the benefit of other librarians' thinking with respect to coordinated national library automation programs.

I might add at this point that we hope for a close liaison also with the Collaborative Library Systems Development project.

The libraries in the Collaborative Systems Project have a higher degree of similarity to each other than do the three national libraries. Our task is complicated by important differences in size and subject specializations. Early in my work as Chairman it became evident that we must examine the present resources and responsibilities of each of the three institutions and the policies and constraints under which each operates in order to search for optimum relationships.

Our study confirmed the conclusion that the three libraries have unique responsibilities involving the collection and dissemination of materials in all languages, in all forms, and from all parts of the world.

The National Agricultural Library has this responsibility for Agriculture and its allied fields, and the National Library of Medicine for the preclinical sciences and for medicine and related fields. The Library of Congress' responsibilities extend to all fields of knowledge, but its cooperative acquisitions arrangements with NAL and NLM, to which I have alluded earlier, defer to those libraries in their special areas of responsibility.

The clientele served by each of the three libraries is similar, with LC having special responsibilities to the Congress, NLM to the medical community, and NAL to the agricultural community.

All serve the general public, although other users may have higher priorities. Each serves other Federal agencies, and each has responsibilities and cooperative arrangements with other libraries, Federal and non-Federal. All have international as well as national service responsibilities.

Services provided by each institution include use of the collections on the premises, interlibrary loan, reference, bibliographic services, publications, photocopying. Each library has varied specialized services related to various user groups.

The common purposes and services indicate that there is sound basis for pressing the quest for a national library system and emphasize the fact that the national libraries of the United States are necessarily the pivot of any true national information system.

A basic ingredient to all systems planning on a network level is, of course, the search for standardization in as many areas of an operation as possible.

Because standardization is such an essential ingredient of any plan to avoid duplication of modules and is an absolute prerequisite for any cooperative system, the Task Force has concentrated attention on the development of standards for the inputting, transmission, and dissemination of information in machine-readable form.

I do not need to talk to this audience on the importance of standards in the new technology or the fact that the usefulness of any standard is proportionate to the extent of its acceptance and use.

All of you know that in any given field the acceptance and use of a national standard is complex and difficult to achieve. The Task Force's experience bears this out. We have reviewed and discussed many drafts, debated many issues, and considered a variety of alternatives before reaching the point where a recommendation for the adoption of the standard is submitted to the three directors.

Thus a great deal of expertise goes into the making of a standard--every concerned person or group must have a voice in the work and every effort must be made to eliminate bias if the result is to be eventual adoption as a nationalstandard.

Despite this lengthy process and the unavoidable backward steps that accompany it, I report with considerable satisfaction some substantial progress.

Of unrivaled importance in standardization and systems development--not only for the three national libraries but for research institutions everywhere--has been the announcement by the directors of the three national libraries of their joint adoption of the Machine-Readable Cataloging format (MARC II) for the communication of bibliographic information in machine-readable form and the set of data elements defined for monographs within the MARC structure.

You have heard on other occasions the history of the development of the MARC format at the Library of Congress in cooperation with other research libraries, so I shall not repeat the account here. MARC reflects the requirements of many institutions, including the three national libraries. It was reviewed by the Task Force and its MARC group in terms of each national library's individual needs.

Adoption has not committed the institution to use all the data elements described; each will determine individual implementation procedures

Agreement on this communications format is a positive demonstration of the three libraries' firm intention to extend the usefulness of their collections and services through the application of new technological capabilities wherever economically and technically feasible, and it will facilitate further extensions throughout the library and research communities.

A second major agreement on standards concerns descriptive cataloging practices and here is where I believe we accomplished what many thought was impossible. We got catalogers together.

In announcing their joint decision to adopt standard practices in descriptive cataloging, the directors emphasized that these standards are of major importance to other libraries, whether manual or computer methods are in use.

Of the common elements identified in descriptive cataloging practices at the three national libraries, six created compatibility problems. To achieve standardization each of the three libraries has agreed to change some practices, and the American Library Association has been asked to make changes in several rules.

I should emphsize at this point that a significant factor in our ability to get at the heart of compatibility problems quickly and to find practical ways of resolving differences in view has been the involvement in the actual work of operating staff from each institution.

The Descriptive Cataloging Group is chaired by NLM's principal cataloging officer; its other members are top cataloging administrators in the other two libraries. Together they were able to come to common agreement on the stumbling blocks to compatibility in their area and on the remedies.

Acceptance of a standard is made appreciably easier if one can assure each director that his principal administrator in that area of specialization has agreed to the proposed practice or change in practice.

A recent further accomplishment has been the adoption by the three national libraries of a standard calendar date code, which is designed to provide a standard way of representing calendar dates in the data processing systems of the national libraries and may be particularly useful for application in data interchange among Federal agencies and among other libraries.

Date in this code will allow for representation of century, year month, or day in the Gregorian calendar. Four digits are provided for use in the computer field to represent pre-twentieth century dates; a six-digit code, based on USASI's proposed code and the Bureau of the Budget standards will represent dates in a field limited exclusively to twentieth century dates.

General use of this standard code will eliminate the confusion caused by a variety of date representations.

I have just received from our Working Groups on Bibliographic Codes a draft standard language code, which I will take up shortly with the Task Force. This code will include languages representing the major body of published literature and has been developed in consideration with language specialists in the three institutions.

A standard character set for use in describing information on magnetic tape is now under final consideration by the Task Force.

The design of this standard has involved consideration of all the characters any of the three national libraries might wish to use to represent bibliographic data in machine-readable form, consideration of the characters that can actually be put into digital form, and the ways in which they can be pulled out once they are in digital form.

The standard set will include some 170 characters, including diacritical marks and scientific characters.

The Task Force is looking into the need for standards that can assure more adequate control over technical report literature. Our Descriptive Cataloging Group is aware of the inadequacy of bibliographical controls over this rising quantity of material and is taking a look at the most feasible avenue for improvement of the situation.

On the basis of a pilot study of the structure of name authority files in each of the three libraries, it has been determined that a mechanized central authority file would be useful. The difference in size of the present files is an important consideration, however, and we await the findings of a larger scale study to provide a factual basis for solid decisions here.

One of the most critical and difficult areas from the point of view of achieving compatibility concerns subject headings, where expressions of both optimism and pessimism have been voiced from time to time. Anyone who has worked in a medical library, as I have, knows what great problems arise in trying to coordinate MESH and the LC subject heading list; right now, of course, this cannot be done. However, we do have a working group looking into this, and there are indications that with some compromises we may be able to achieve some success here.

The study group in this area has tackled the issues in a most constructive way. Sub-headings in use in each institution are being explored, charts showing the interrelationships of the headings in use have been drawn, and the possibility of establishing a common list has at least been aired.

Computer output programs useful to the three institutions are being examined, including collective publications, on-line and off-line printing, and console output.

Inasmuch as initial objectives set by the directors included the creation of a national data base for serial publications, the progress of the National Serials Data Program has had a high priority. I am assuming that all of you are well acquainted with this ambitious undertaking, supported jointly by the three national libraries together with funding from the National Science Foundation and the Council on Library Resources, Inc. I will there fore omit the details of the work that has occupied a sizeable amount of the time of our technical people the past year, resulting in the compilation of data elements required for the control of serials, now under consideration by the three libraries.

Although much more work and many more resources will have to be poured into the program, the ultimate product will be a matchless tool for the bibliographic control of the millions of pieces of serial literature coming into this country from all parts of the world.

We have learned some interesting facts from the serials work to date: first, a machine-based national data bank should be designed to take maximum advantage of computer systems and should not be constrained by the limitations of manual systems. Second, a universal numbering scheme for serials is a basic requirement--the Task Force has been cooperating with USASI's Z-39 Committee in an effort to get a proposed scheme underway here--and third, users' attitudes on implementation are so in variance that it is not likely that the final recommendations will satisfy everyone. But since they seldom do, we are determined not to retreat from our original ultimate purpose of developing a communications format for serials comparable to the MARC format for monographs.

I do not need to elaborate on the reasons why this assignment is far more difficult than the development of MARC I and II. All of you recognize that in MARC we had the standard printed card as a beginning; with serials we have lacked this standardization, and it is this initial task that has taken the concentrated attention of the staff in the first phase of this vital program.

This compatibility and standardization are absolutely essential to any kind of systems approach. Since last December the Task Force and its Systems Group have been agonizing over the matter of alternate systems and design possibilities. There has been progress in analysis of present methods of each library, and this is continuing.

Our systems work has been handicapped by the lack of a sufficient number of trained people who can devote full-time to the necessary detailed studies for an extended period. The Council on Library Resources has generously assigned a systems analyst to the Task Force, but other Working Group members are carrying additional responsibilites. While the Task Force has made a number of attempts to solve this problem, the difficulty of finding staff with the necessary and unusual combination of computer orientation and librarianship is well known.

There is the further complication that at least for the next few years there will necessarily have to be three discrete systems; our interim objective, therefore, must be to find appropriate ways to build bridges between these systems and to continue to plan ahead for a later time when a more ideal system can be visualized, with a central switching mechanism that will provide access to the total knowledge contained in the three libraries.

Each of the three national libraries is presently committed to automation programs that make it necessary for our systems specialists to plan for appropriate interchange and linking of these systems.

Our Systems Working Group is now attacking this "short-range" planning which involves the coordination of the present systems design work at the three institutions, the identification and planning for the actual interchange of system modules if and where appropriate, and the interconnecting of the three discrete systems.

The NLM and NAL systems are planned to become operational in the early 1970's and to continue probably through much of that decade. LC's approach probably will call for certain segments to become operational in the early 1970's and to continue at least into the 1980's. Selected segments of each system may be available prior to these periods and may be in use beyond these general time frames.

We do know that because of size, the NLM and NAL systems can be expected to become operational at an earlier date than that at LC, although, through the modular approach, LC will have sub-systems being phased into operation ahead of the total capability. It is with these advanced subsystems at LC that the NAL and NLM systems will be interlinked in the short-range plan. The planning for the interlinking of these systems is necessarily constrained by existing organizational structures of the three libraries, by normal technical constraints, and by their respective assigned missions.

Thus, considerations to date appear to point toward the concept of three data stores mutually capable of receiving and transmitting information. This would mean that each library would create its own store of machine-readable information, with each store having the capability of receiving data from the other two libraries and of transmitting data to them.

Because of the overlap in many fields of knowledge today, because modern science and modern scholarship are so interdisciplinary, it will be necessary to create a situation that can provide for the economical dissemination of information to any community needing it from any repository holding it. We can suppose that the information will come in raw form to the three national libraries, where it will be digested by each library and made available in different forms for different clientele. If the methods by which we digest and store the information are compatible, then we will be able to make it easily accessible from any store in which it is located.

Beyond this we are also faced with the need for long-range in-depth planning for the period beyond the 1970's when there might be more freedom to search for optimum interactions. Such long-range planning must include reexamination of the three national libraries' goals and objectives for the long-term system and consideration of the possibilities for combining functions and integrating certain operations as appropriate. We recognize that it is impossible to continue "as is."

It is essential to pursue the planning for both the short- and long-range time periods simultaneously. Because the possibilities are so far-reaching, this long-range study whould begin as soon as possible, and I have been pressing for the search for funds and personnel to make some substantial progress here. The ultimate decisions will remain with the three directors, of course.

Among the Task Force's targets for the months ahead, in addition to acceleration of the National Serials Data Program, cooperation with the Z-39 Committee on the universal numbering scheme for serials, and continued work on standards and compatibility problems, are considerations on the assignment of responsibilities in a national network. This is an essential ingredient of the long-range planning.

It is my firm conviction that any effective system must be based on the principle of elimination of duplication of effort. There is too much to be done in this total area and too many demands on limited resources of talent and money to allow duplication of each other's work. Ideally it would seem logical to allocate sole responsibility for specific functions in specific subject fields to one institution, and the Task Force has had some illuminating discussions of alternative possibilities along these lines, particularly in connection with acquisitions and processing functions.

Again, these are questions that do not lend themselves to easy resolution because of the specific responsibilities assigned by statute to a particular library, because of the historical development of individual policies and special relationships, and because of the special competences within the individual libraries for particular functions.

Nevertheless, the Task Force intends to continue its look at possible new patterns that in time might prove useful, economical, and acceptable to all concerned. We are convinced that the time is long overdue when the three national libraries, with their combined holdings representing almost the total of recorded knowledge, can lead the way to a new and exciting era of interlibrary cooperation, both national and international. The directors, in launching this program, have recognized that if we can unite in working out new and more effective and rapid ways of operating our complex apparatus then we will all respond with more awareness and efficiency to the needs of the total research community.

It is too early yet to foresee all the implications this effort can have upon the library community at large. Certainly the adoption of MARC II as a standard for providing catalog information in machine-readable form increases substantially the versatility of its use in libraries because the computer, as we all know, permits a greater variety of approaches to the information than card or book catalogs and because libraries with computer facilities can print out more easily and quickly a greater variety of research tools.

All of the standards we have developed thus far will benefit other libraries desiring to automate, and there is promise that through increased collaboration and sharing of knowledge and resources our common problems can be alleviated more quickly and, hopefully, more economically.

It will all take time, There are no easy paths, and much of the work must be a pioneer effort. The Collaborative Library Systems Development program and the U.S. National Libraries Task Force can cooperate through the sharing of information and specific results of their respective studies. There are a number of ways in which our cooperation can be augmented. Joint meetings at appropriate times could provide valuable give-and-take at the working level. A collective "Skills Bank" might widen the use that we could all make of the scarce and absolutely essential talents of trained systems staff. Directors of all the libraries involved in the two programs might profit from a creative colloquium on a collaborative systems network.

Before closing I want to stress the remarkable achievement that has been realized by the commitment of the Library of Congress, the National Agricultural Library, and the National Library of Medicine to work together in a cooperative enterprise of this magnitude. It is without doubt the largest effort, in terms of talent and man-hours expended, toward national library cooperation that these libraries have ever undertaken. The decision to join together in this effort will have far-reaching results in the long run for librarians and scholars in future generations.

The excellence of American libraries over the years has rested in large measure upon the extent to which cooperative enterprises have been successfully undertaken. We believe that the Task Force's program gives conspicuous evidence of the fact that collaborative effort at the real working level offers the best chance of finding durable settlements to crucial library questions. We hope that our effort will be contagious, and we invite all interested librarians to give us their help and their support.

Discussion

Rogers: I'll ask Sam to stay on the podium. If you want to address questions to him, do so directly.

Lazerow: I have to warn my colleagues in the audience, whom I will name, that I will occasionally throw some questions to them: Paul Reimers, Information Systems Coordinator, Library of Congress; Ralph Simmons, a member of the Task Force; and Mr. William J. Welsh, Director, LC's Processing Department.

Kilgour: Is the document on the compatability of descriptive cataloging available?

Lazerow: I knew you would ask that question. There is no actual report. There is a document on descriptive cataloging which is the recommendation (Recommendation Number 3) of the Task Force summitted to the three directors on the compatability in descriptive cataloging. It has not yet been published, but there's no objection to making it available to anybody who wants it.

Kilgour: This really is in congratulations and gratitude to Sam and the directors of the National Libraries. They have gotten over an enormous hurdle, and produced what amounts to a large accomplishment. Six years ago, when Sam and I were both in medical libraries, we talked about related matters and our tone of voice certainly reflected the fact that it probably would never happen. But it has happened; somebody ought to say thank you, and I do on the behalf of all of us here. We're terribly grateful, Sam.

Veaner: I'd like to ask a question about the machine readable authority files: do you see the authority files for personal names and for corporate bodies as two separable entities that would be handled, in a technical sense, as separate problems, or as the same? Would they be subject to the same file creation rules?

Lazerow: This is a problem the working group has not yet gone into. What the working group has done is to take a hundred entries from each of the smaller libraries, NAL and NLM, and run them against the file of the larger library, LC, to find out where there were conflicts, why they weren't used the same way, and so forth. My own feeling is that I don't see why they have to be handled separately in a computer store. Perhaps Mr. Welsh would like to comment.

Welsh: I don't see any particular need to separate them. We can for purposes of the record indicate whether the authority is a personal name or a corporate body. We're looking at this from the main entry point of view. We can add subfields to identify parts of the entry. We will have a separate file for subjects. Is there something more to your question that escapes me?

Veaner: Maybe I ought to defer to Ed Parker in this regard. We have wondered whether it would not be useful in the future to have some kind of directory of corporate bodies very similar to a national directory of serial titles which the National Serials Data Program is considering. We wondered whether such a concept would have any implications for the Task Force's work, or what their views were on this matter.

Welsh: Well, if this is desirable there's no reason why it can't be done. I say you can identify the authority file as being a corporate body and you can spin off a directory or listing of the corporate bodies. You might need this for other purposes.

Veaner: But you see no intrinsic distinctions for setting up the authority file?

Welsh: I do not.

Fussler: Could you comment on the Task Force efforts to date or plans with respect to handling non-Roman, alphabetic, bibliographical data in machine readable systems?

Lazerow: We have not gone into the non-Roman materials yet. I think Mr. Reimers is probably more familiar with the situation than I am. The character set which I described deals only with the Roman alphabet. Paul, would you care to comment on that?

Reimers: The question here is again one of access. How are you going to access these items? I think that we can look forward in the not too distant future to the Orientalist, who puts in the logographs; we could input these into a computer in some kind of digital form. But how do we address this? This, I think, is our real problem if the problem again gets back to problems of use. Fred Kilgour tells me I haven't looked at the real user of serials in regard to the National Serials Data Program. I think here we have to look at the user too. Since we are dealing with a computer, we are not concerned with filing rules as such. We're not concerned with interfiling so much as we are with how people are going to access this file. Are the Oriental scholars actually going to draw the characters on the face of the cathode ray tube with a light pen in order to access a record? I don't think anyone in this room is really going to see this in terms of computer technology, because this gets back to the semantic problems upon which automatic translation floundered some five or six years ago. I think we're going to have to depend on standard forms of transliteration.

Lazerow: I think that the crux of the problem is that we have enough trouble working with ABC's. This is what we want to solve first, before we get involved with these other things.

Hammer: I'd like to ask a question of Paul Fasana and Allen Veaner if I may, from this morning's earlier talks. Mr. Fussler gave information on the resources going into his part of the project at Chicago. I wonder if we could get the same information if it's available for Columbia and Stanford, in terms of people and money.

Fasana: As long as you won't hold me to the accuracy of these figures. Our Library Systems Development Project has a grant of $350,000, $200,000 of which is supplied by NSF, and $125,000 is in house money. This is for an eighteen month period. The CLSD project described by Dr. Logsdon is $60,700 over an eighteen month period also. Our Reserves Project, funded by the Office of Education, was a $90,000 project over an eighteen month period of time.

All other work excepting work in PIC and Upper Mantle is sponsored or funded with in-house funds. I don't have the figures for PIC because we don't handle those budgets. In terms of Systems Office personnel there are four full time "computer types:" one senior systems programmer, one junior systems programmer, plus two regular programmers. In addition there are three to six library systems analysts. These are essentially librarians who have been trained in systems work who spend anywhere from 25% to 100% of their time doing the systems work. In addition there is input keying staff of about six. The PIC project has an additional four or five clerical personnel, plus eight to ten professionals doing indexing, descriptive cataloging, etc. The Upper Mantle Project has three to roughly four people of which one is a professional librarian.

Veaner: We have two projects working collaboratively on a local basis, so it is somewhat difficult at times to assess just how many people are working, because the number tends to change from time to time. In the library project, Stanford has six full time persons, four of whom are systems analysts, two of whom are librarians. One of the librarians is a research assistant. We have an additional person just recently hired who is a data control supervisor. In software development, working under Ed Parker's direction, we have a great deal of Ed's own time, a full time Programming Manager, Dick Bielsker, and about five or six full time programmers.

Bielsker: We must add to this number a considerable amount of expertise that we have on call from the staff of the Computation Center, ranging from the Associate Director of the Center himself, who is responsible for the largest facility on campus, as well as other systems programmers and graduate students. I hope that accurately describes our staffing. Our grant from the Office of Education, is $417,000 for an eighteen month period, to which Stanford is adding a susbstantial amount in cost sharing. Ed Parker's grant is from NSF; would you care to comment on that, Ed?

Parker: The latest funding was $274,000 for twelve months starting the first of July of this year. There was a previous period of 18 months at a lower rate of funding.

Veaner: I think that summarizes our situation.

Lazerow: I'd like to add one thing. The National Libraries investment might be of interest to some of you. The people involved in this effort are all staff members of the national libraries except one person supported by the Council on Library Resources. They are involved in these ten working groups, as staff assistants, in doing the special studies that are necessary. There are eighty-five memberships involved, representing sixty individuals. There have been something like a hundred and fifty meetings of Task Force and Working Groups since the early part of this year, and I would estimate that there has been something like thirty thousand man-hours or work invested in this collaborative effort, which is roughly the equivalent of fifteen man years. In addition, the Serials Data Progaam was financed in Phase One to the extent of $130,000 by the National Science Foundation, the Council on Library Resources, and the three national libraries. For this fiscal year, the Library of Congress intends to shoulder the entire cost itself of Phase Two.

Management of the Design and Development of the Biomedical Communications Network

Ralph A. Simmons
Head, Information and Computer Sciences
National Library of Medicine, Bethesda

A Paper Prepared for the Stanford Conference on Collaborative Library Systems Development, October 4-5, 1968

On August the 3rd, 1968 the 90th Congress passed a joint resolution authorizing the establishment of a National Center for Biomedical Communications and designating it as the Lister Hill National Center for Biomedical Communications. The Center has been endorsed by the Scientific Community as an urgently required facility for the improvement of communications so necessary to health education, research and practice and established as a part of the National Library of Medicine. Its designation as the Lister Hill Center was as a tribute to the career of Senator Hill of Alabama, who has accomplished so much for the health of the American people.

There have been many significant activities and trends in the past few years that have led to the need for this National Center. The Federal Government has played an ever increasing role in the provision of health services and in the development and conduct of medical research and educational programs. The establishment of the Regional Medical Program under Federal sponsorship and direction through the National Institutes of Health represents a milestone in the organization of national resources toward the improvement of the nation's health. The Veteran's Administration is assuming an expanding role through a variety of programs for the improvement of health and health care. All of this concentrated effort is in response to the demands of society for ever-improving health care and prevention of sickness. It also represents a principle of decentralization of operations of the responsive programs and a centralization of supporting resource allocation. Further impetus to the establishment of the Center has come from the national attention to networks and communications as the way to the improvement of the necessary transfer of knowledge to support the variety of expanding medical programs. It also represents a response to the need for the improvement in the coordination of technology development and application in the areas of information and computer sciences.

---

* p. 88

The principal responsibilities of the Lister Hill Center for Biomedical Communications are described under these four major functions: 1. the design, development, implementation and management of the Biomedical Communications Network; 2. the application of existing and advanced technology to the improvement of biomedical communications; 3. to serve as the focal point in the Department of Health, Education and Welfare for the technological aspects of biomedical communications, information systems, and network projects; and 4. to represent the Department in the activities of the President's Office of Science and Technology, other Federal agencies and interagency committees in areas related to information and communications. It is the first of these functions -- the establishment and operation of the Biomedical Communications Network -- that is the principal concern of my following comments.

Why a network at all? What are the advantages for biomedical information services to be gained through networking? These can best be expressed by these five conditions that represent needs for such a network: 1. the existence of a unique collection in a single location that is useful to a dispersed audience; 2. the inadequacy of local collections and the need for complementary support from other sources; 3. the centralization of particular capabilities or unusual resources with a dispersed need; 4. the need for interpersonnel, direct communication; and 5. the justification for the distribution of certain responsibilities among organizations or regions based upon economic or professional capabilities. The linking of libraries, information centers, medical schools, hospitals and research centers through communications arranged so as to constitute a network can best meet those needs and conditions as described.

The selection of a network for improving the information and educational services within the medical community was also based upon the present state-of-the-art in information and computer sciences. The network when looked upon as a complex process including communications, controls, and feedback and consisting of a variety of components is at the proper step in a "complexity" ladder of technological advancement. We have passed through the stages of the use of the individual computer and then computer systems and now see extensive efforts in the linking of the computer systems into networks. Major research and development is now underway on the next step on the ladder -- automatons and the mechanization of intelligent behavior. When you have R&D at the next stage, you know that you are in the right stage on the ladder of technical complexity for the development of an operating activity.

The specific objectives of the Biomedical Communications Network (BCN) against which we can test each stage of our development effort are five in number: 1. to improve research; 2. to provide better professional services; 3. to make conscious and planned decisions on the applications of technologies to biomedical communication; 4. to provide for a more uniform, highly-qualified professional; and 5. to provide for a larger, well-informed citizen audience. A fundamental concept that information systems in themselves are a completely sterile and artificial resource and that they must be coupled with some process forms an additional guide to the establishment of the BCN. In this case, the process with which we must couple the network is that of medical education. This is not surprising if we consider that an important purpose of medical education is the transfer of skills, knowledge, and information from a variety of sources through a variety of media to the student and practitioner.

The characteristics of our network can be expressed as determined by the customer requirements. The various services of the network will be available on a decentralized basis and accessible through local hospitals, medical societies, clinics, medical schools, medical libraries, and private offices. These services will be organized along the lines of topical specialties and against the major medical advances accomplished in the latest five years.

The planning to date for the BCN has included the division of the Network into five major component parts, i.e., the Library Component, the Specialized Information Services Component, the Specialized Educational Services Component, the Audio and Audiovisual Services Component, and the supporting Data Processing and Data Transmission Facilities. Our major concern today is the Library Component but, before examining it in some detail, I would like to define the scope of the other elements. The purpose of the Specialized Information Services Component is to communicate information related to specific subject areas to customers in the bio-medical and health-related fields using communication, computer, and other relevant technologies. Its principal constituents are planned to be a referral center, a distinct toxicological information system and a system of information analysis centers.

The Specialized Educational Services Component has as its goal the support of three distinct areas of education: 1. continuing medical education for the medical professional; 2. education of the medically uninformed; and 3. education in related relevant technology for the medical professional such as new devices, new communication media, or new procedures. As the names imply, the Audio and Audiovisual Services Component provides identification of available materials and access to those materials and the Data Processing and Data Transmission Facilities provide the support in the identified areas as required.

The Library Component of the BCN is intended to provide bibliographic citations to biomedical literature, access to the literature itself, and support to the required library operations in such areas as acquisitions, cataloging, indexing, and announcement and reference services. I realize that networks to librarians are really nothing new. The interlibrary loan activities among libraries have demonstrated networking on a regional and national scale. The complex systems of national and regional bibliographic control in the form of union lists and catalogs and the systems of interim source referral services clearly complete the identification of the library system as a viable _de facto_ network. But with the newer tools provided by our advancing technology, the network takes on a completely new dimension. It is the planning for the development and management of this more advance network that I now wish to discuss in some detail.

Actions on the part of the staff of NLM and others in the medical library profession over the past few years, supported by specific legislation, have resulted in the establishment of the nucleus of a biomedical library network including Regional Medical Libraries, decentralized MEDLARS Centers, and affiliates in England and Sweden. Under the Medical Library Assistance Act of 1965 regional medical libraries have been established through Federal funding at

Harvard, the University of Washington, the College of Physicians in Philadelphia, the John Crerar Library in Chicago and, soon to be added, Wayne State University and the New York Academy of Medicine. These institutions, as you know, have received grants from the National Library of Medicine to provide specific services to their respective regions. In addition, NLM has contracted with a series of institutions to provide specialized MEDLARS services for customers located in their respective areas. These activities, known as decentralized MEDLARS centers, are at Harvard, Colorado, UCLA, Alabama, Ohio State, and Michigan. The services include the formulation of literature citation searches on local computers or the transmittal of the searches to NLM to be run there. Affiliated MEDLARS Centers are also in operation at the National Lending Library in England and at the Karolinska Institute in Sweden.

The future Library Services Component of the BCN is to be built upon this beginning. It is expected to add to the numbers of regional medical libraries and to the decentralized MEDLARS Centers, to include the various systems of Federal medical libraries, to extend to all university medical libraries and networks, and to reach the individual hospital and other health science libraries. These organizations will be grouped under the Library Component in four basic levels of network participation -- as shown on this chart*-- and the levels will be principally determined by the access provided at each to the various data bases to be included. These levels are as follows:

Level 1 - The Lister Hill National Center for Biomedical Communications.

The National Library of Medicine is to serve as the hub of the BCN and of its Library Services Component. It will include the major input processing for the construction of the bibliographic data bases, i.e., MEDLARS and Current Catalog files, with input support from other levels as appropriate. The network control and management will be exercised from the Center and the major data bases will be accessible from on-line machine storage. Major computer and communications facilities will support this Center.

Level 2 - Decentralized MEDLARS Centers/Regional Medical Libraries.

This second level of the network will be characterized by major computer facilities providing on-line access to the majority of the data bases located at the Lister Hill Center. This access is to be through communications links to the central files at NLM or by the placement of these files in on-line computer storage at the secondary centers themselves. There will also exist communications links among the level 2 nodes in what can be termed a horizontal pattern.

---

* p. 103

Level 3 - Regional BCN Access Centers.

The third level nodes are to be terminal access centers with input/output devices and communications equipment permitting the transmission of alpha numeric data between these centers and the Lister Hill Center and/or the Level 2 centers. The communications with the two higher levels of the network will be with the computer files at those levels and with communications terminal devices for simple message transmission. Links will also be provided among the fifty to seventy-five access centers comprising this level.

Level 4 - Local BCN Terminals.

The fourth, and last, level in the network will include 150 to 200 local terminals consisting of input/output equipment for the exchange of alpha numeric data with any of the nodes in the other three levels and among those in the fourth level itself. This exchange will be only data transmission from communications terminal to communications terminal and will not permit linking directly to a computer file.

An essential part of the network planning effort is to identify other related activities and to build the proper interactions with these activities. The four major communities are shown on this chart*and there are also listed a sampling of specific activities that will have an impact on, or will be affected by, the Library Services Component.

Our program at NLM for the development of the Biomedical Communications Network is under the direction of Dr. Ruth Davis who is the Associate Director for Research and Development and who also has been named as the Director of the newly-created Lister Hill National Center. It is her belief, and that of those of us on her staff, that the development of the BCN as a service-oriented mechanism demands effective and formalized management policy and proce-

---

* p. 104

dures which we have chosen to call The BCN Management Process. Such management processes have been shown to be critical to successful network, or system, implementation during the ten to fifteen year history of system design work. There is a rather extensive body of documentation --principally report literature -- that has grown up around the subject of system design. An oversimplified and yet very useful review of the elements of system design can be gained from their arrangement in this three-part list.* Formalized management procedures must be followed to ensure attention to these elements and to provide adequate control and direction during the entire network design and implementation cycle. There is no question but that effective management has become a pacing element in all applications of technology. In addition, management provides the means of accomodating to the rapid pace of technological development, the complexity of networks, the diversity of organizations involved and the frequent and unavoidable changes in requirements. Effective management is in essence equivalent to an orderly approach to a problem. The steps to be followed in the solution of the problem can be listed in many ways.* Those shown on this chart can be recognized as most frequently used in relationship to the solution of a scientific problem such as in biology or chemistry. They can also be used, however, as the outline to be followed for the solution of management problems and form the basis for the approach known as scientific management. It is this approach to management which provides the necessary stability and continuity to maximize the performance of individuals involved in the system process.

The purposes of the BCN Management Process are: 1. to delineate the requirements, policies and procedures for the conceptual, definition, design, development, acquisition and initial operational phases of the program and 2. to prescribe the significant management actions for integrating and fulfilling the responsibilities of the organizational elements involved.

The objectives of this Management Process can be clearly identified. They are:

1. To ensure effective management throughout the network cycle. For the BCN, the cycle is comprised of the conceptual, definitive,

---

*p. 105
*p. 106

design, development, acquisition, and initial operational phases.

2. To balance the factors of performance, time, cost, and other resources to obtain the BCN. This objective involves the preparation of the necessary budget submissions, related programming and planning data, funding documents, and resource allocation schedules. It permits the assignment of priorities and either precludes schedule slippages or prevents surprises in such slippages.

3. To minimize technical, economic, and schedule risks.

4. To control changes to requirements so as to minimize slippages and ensure maximum utilization of work completed or underway.

5. To provide documentation supporting decisions made and actions taken.

6. To establish a discipline, or blueprint, for the Lister Hill Center staff to follow so that the coordination of planning and action is maintained between the management officials responsible for the various phases of the network cycle.

7. To manage and control contractor efforts.

8. To identify and schedule significant actions to be accomplished and to effect their accomplishment.

9. To establish requirements for the flow of information between the responsible managers and organizational elements.

10. To undertake the research and development efforts necessary for the BCN.

The customers or the user communities associated with each of the BCN components can be separately treated during the early stages of the BCN development cycle. This is not due to their disparate composition but rather to the disparate nature of the services or products offered by the various BCN components. Although one of the distinguishing characteristics of the BCN is its unification of education and information resources for maximum benefit to individual customers, the nature of this unification does not derive primarily from the customer. Rather, the BCN management staff must generate feasible and alternative means of effecting unification of product so that selection of the appropriate means can be consciously made by responsible authorities and users. This separation of customer group and services by BCN component should be recognized as an essential feature which has permitted parallel but separate efforts to be undertaken for each component. The unifying effort is the responsibility of the Lister Hill Center staff.

The management process for the network program is dependent upon adequate documentation which defines both the BCN and the management process itself. The generation of this documentation is an essential element of the management process. A review of the "system" literature has allowed the staff to select a minimum but critical set of documents which will be necessary for the BCN implementation.

Management must be presented with a logical and complete array of information if the required decisions are to be made, from that to undertake the design in the first place to the more specific directions as to the specifications for operation and the acceptability of the design itself. The planning must be laid out in an orderly fashion to provide this required information to managers, to the designers, and to those who are to implement the system after acceptance. The environment within which the system must operate and the expression of need for the operation being developed must be clearly outlined and, with management endorsement, constitute the justification and statement of objectives for the entire project. From this point, the planning descends through a chain of increasing detail in the expression of objectives, performance requirements, description of system to meet the needs, engineering design of the system, and provision for test and evaluation of the resulting system. Accompanying this design planning must be the management plan that describes the responsibilities for developing, testing, evaluating and operating actions and assigns these responsibilities. It also includes resource and work schedules to be used as controls for the entire project.

The documentation set selected to meet these demands includes four distinct elements: 1. the Statement of Requirements; 2. the Technical Development Plan; 3. the Network Engineering Plan; and 4. the Network Management Plan. The purpose of each is briefly stated on this slide. This set is considered critical enough to the entire process to warrant further detailed examination.

The Statement of Requirements includes both a description of the general requirements for the network, or its components, and the specific operational requirements. It discusses the needs of society for improved communication of biomedical knowledge and defines these needs against the total background of all possible customers in our society. It also presents the basic philosophy and concepts for the BCN as dictated by the expressed needs. This first major segment of the document series must set the stage for all later efforts by placing those efforts in the context of the total community and providing the expression of the basic mission and/or objectives of the total project.

Within this first document of the series there must also be included the next level of planning -- the specific operational requirements. These must be established from the general objectives previously defined and must delineate and/or define the following series of activities or facts:

1. The services and products to be provided by the BCN to meet the needs of the users;
2. The functions and operations to be performed in order to produce these services and products; and
3. The characteristics of the customers in order to ascertain the match of users against the designated services and products.

The general services and products must be further defined in terms of such parameters as quality, quantity, timeliness, reliability, accessibility, and format. The orderly and systematic presentation of the general and specific operational requirements as outlined permits one to proceed to the development of the technical specifications and constraints for the Network and its components.

The Technical Development Plan (TDP) translates the statement of requirements into a coherent description of a network which, when operational, will satisfy the users' needs. The TDP is the bridge between the intended users of the network and the engineers and technicians who will direct the design and development of the network; it defines the operating environment and prescribes the general parameters of the network. It provides the foundation on which system engineers can postulate detailed network designs, formulate operating specifications, identify specific development tasks, set schedules, and estimate detailed resource requirements. The outline for the Technical Development Plan for the BCN is as shown on this slide.*

The next, and third, document in the set is the Network Engineering Plan. It covers the system efforts which normally begin after the network requirements have been established and continue until an operating system is accepted by management. The Engineering Plan covers system definition and system design. It builds upon and refines system requirements previously defined so that they can be translated into design requirements. A series of iterations is outlined which being with gross tradeoffs among cost, performance, and schedules and proceed toward final system definition and implementation. Previous studies which affect system design are reviewed and documented as part of the Engineering Plan. Several design approaches are examined and evaluated, based upon proper considerations of variables such as services to be performed, facilities, communication, computer programs, procedural data, training, testing and evaluation, logistics, and intrasystem and intersystem interfaces. In this process, major technological problems which would cause unacceptable delays are eliminated. As the process continues, performance requirements and constraints are documented. The objective of the Network Engineering Plan is to make possible the selection of the best design approach from the alternatives examined and then ensure that the desired system is designed at the least possible cost.

The presentation of a formalized structure of management efforts to establish and maintain positive management control of the progress of the BCN development is contained in the Network Management Plan. The basic ingredient of this Plan is a road map defining the major management actions to be accomplished during each phase of the Network development. It is to provide all levels of the participating management with a common understanding of the administrative, financial, logistical, and other supporting factors which are essential to the implementation of the BCN project. The Plan shows the interrelationships of the

* p. 110

Network components and outlines the Network interface with other systems. It identifies and contains information on: a program summary, schedules, program management, operations, manpower, organization, finance, and work authorizations. The Network Management Plan is a tool for project control and direction. It provides a systematic way for the Project Director to make intelligent judgements on resource allocation and phasing of project activities.

This final slide presents the entire BCN Management Process on a single chart.* It is divided into the four major phases identified as conceptual; definition; design, development and acquisition; and operational. Work is currently taking place simultaneously in all four phases as dictated by the conditions in the real world situation demanding service now. It is not possible to proceed as would be theoretically desirable completing each phase before moving on to the next.

As with any activity, the success of this BCN Management Process depends upon clearly defined lines of responsibility for the accomplishment of each phase of the Process and for each element within a phase. The staff of the Lister Hill National Center for Biomedical Communication has the responsibility for the overall process and is supported by other elements of the National Library of Medicine who have been given roles of responsibility, approval or coordination in specific functional areas of the process.

*p. 112

Public Law 90-456
90th Congress, S. J. Res. 193
August 3, 1968

JOINT RESOLUTION

To designate the National Center for Biomedical Communications
The LISTER HILL NATIONAL CENTER for
Biomedical Communications

....The Lister Hill Biomedical Communications Center to be constructed and located as part of the National Library of Medicine....

....This center strongly indorsed by representatives of the Scientific Community as an urgently required facility for the improvement of communications necessary to: Health education, research, and practice....

....This center would function to contribute to lifelong objectives of Senator Lister Hill's legislative career....

APPROVED August 3, 1968

Congressional Record Vol. 114 (1968)
July 19: Considered and passed Senate
July 24: Considered and passed House.

LISTER HILL NATIONAL CENTER FOR BIOMEDICAL COMMUNICATIONS

Major activities related to the BCN

- Federal role re health services changing
- Federal role re medical research and education changing
- RMP recently underway
- VA assuming expanding role
- Added demands for maintenance of medical excellence
- Decentralization of operations and centralization of resource allocation
- National attention re networks and communications
- Improved coordination of technology development

LISTER HILL NATIONAL CENTER FOR BIOMEDICAL COMMUNICATIONS

FUNCTIONS

- Design, development, implementation and management of a Biomedical Communications Network.

- Application of existing and advanced technology to the improvement of biomedical communications.

- Focal point in DHEW for technological aspects of biomedical communications, information systems and network projects.

- Representation of the Department of Health, Education, and Welfare in the Office of Science and Technology, other federal agencies and interagency activities.

NEED FOR NETWORKS

- Unique collection at single location useful to dispersed audience
- Inadequacy of local collections and need for complementary support
- Centralization of capabilities or resources with dispersed need
- Need for interpersonal direct communication
- Economic or professional justification for distribution of responsibilities among organizations or regions

A NETWORK AS A CHOSEN INSTRUMENT

- Complex process providing:

  - Communication
  - Control
  - Feedback
  - Variety of components

- Appears at proper step of complexity in terms of state-of-the-art

Decreasing Complexity ↓

5 Assemblies of systems, networks, automatons
4 Automaton
3 Network
2 System
1 Individual equipment device

BIOMEDICAL COMMUNICATIONS NETWORK

Characteristics as determined by customer requirements

- Services available on decentralized basis

  Access through local
    - Hospitals
    - Medical societies
    - Clinics
    - Medical schools
    - Medical libraries
    - Private offices

- Services organized
    - Along topical specialty lines
    - Against advances in latest five years

NETWORK MANAGEMENT PLAN

Presents Structure of Management Efforts for BCN Development

- Defines management actions for each phase of BCN development

- Documents Administrative, Financial, Logistical, other factors essential to implementation of BCN Project, including details on:

  - Program summary
  - Schedules
  - Program management
  - Operations
  - Manpower
  - Organization
  - Finance
  - Contracts

It is a tool for Project Control & Direction

SPECIALIZED INFORMATION SERVICES
COMPONENT

Communicate information related to specific subject areas to customers in the biomedical and health-related fields using communication, computer and other relevant technologies.

COMPONENTS OF THE BIOMEDICAL COMMUNICATIONS NETWORK

Library Services Component

Specialized Information Services Component

Specialized Educational Services Component

Audio and Audio-visual Services Component

Data Processing and Data Transmission Facilities

SPECIALIZED INFORMATION SERVICES COMPONENT

CONSTITUENTS

- Referral Center
- Toxicological Information System
- System of Information Analysis Centers
    - PHS
    - Other Agencies

SPECIALIZED EDUCATIONAL SERVICES
COMPONENT OF THE
BIOMEDICAL COMMUNICATIONS NETWORK

- Continuing medical education for the medical professional
  - Protection of the trained adult from technical obsolescence
- Education of the medically uninformed
  - Special topical areas for the educated
- Education in related relevant technology for the medical professional
  - New devices
  - New communication media
  - New procedures

LIBRARY COMPONENT
OF THE
BIOMEDICAL COMMUNICATIONS NETWORK

Objective is to provide:

- Bibliographic citations to biomedical literature
- Access to the literature
  (Document or copy thereof)
- Support to library operations
  - Acquisitions
  - Cataloging
  - Indexing
  - Announcement services
  - Reference services

ELEMENTS OF THE LIBRARY COMPONENT

-- Regional Medical Libraries

- Harvard
- University of Washington
- College of Physicians, Philadelphia
- Wayne State University, Detroit
- John Crerar Library, Chicago
- New York Academy of Medicine

-- Decentralized MEDLARS Centers

- Harvard
- Colorado
- UCLA
- Alabama
- Ohio State
- Michigan

-- Affiliated Centers in England and Sweden

ELEMENTS OF THE LIBRARY COMPONENT

FUTURE

- -- Regional Medical Libraries
  - . Five additional libraries under consideration

- -- Decentralized MEDLARS Centers
  - . Expansion to total of 10-15

- -- Federal Medical Libraries
  - . VA, Military Services, etc.

- -- University Medical Libraries and Networks
  - . SUNY, etc.

- -- Hospital and other Health Science Libraries

CHARACTERISTICS OF NETWORK NODES

Level 1 - CBC
- Input processing for construction of data bases
- Network control and management
- Major computer and communications facilities
- Data bases accessible on-line

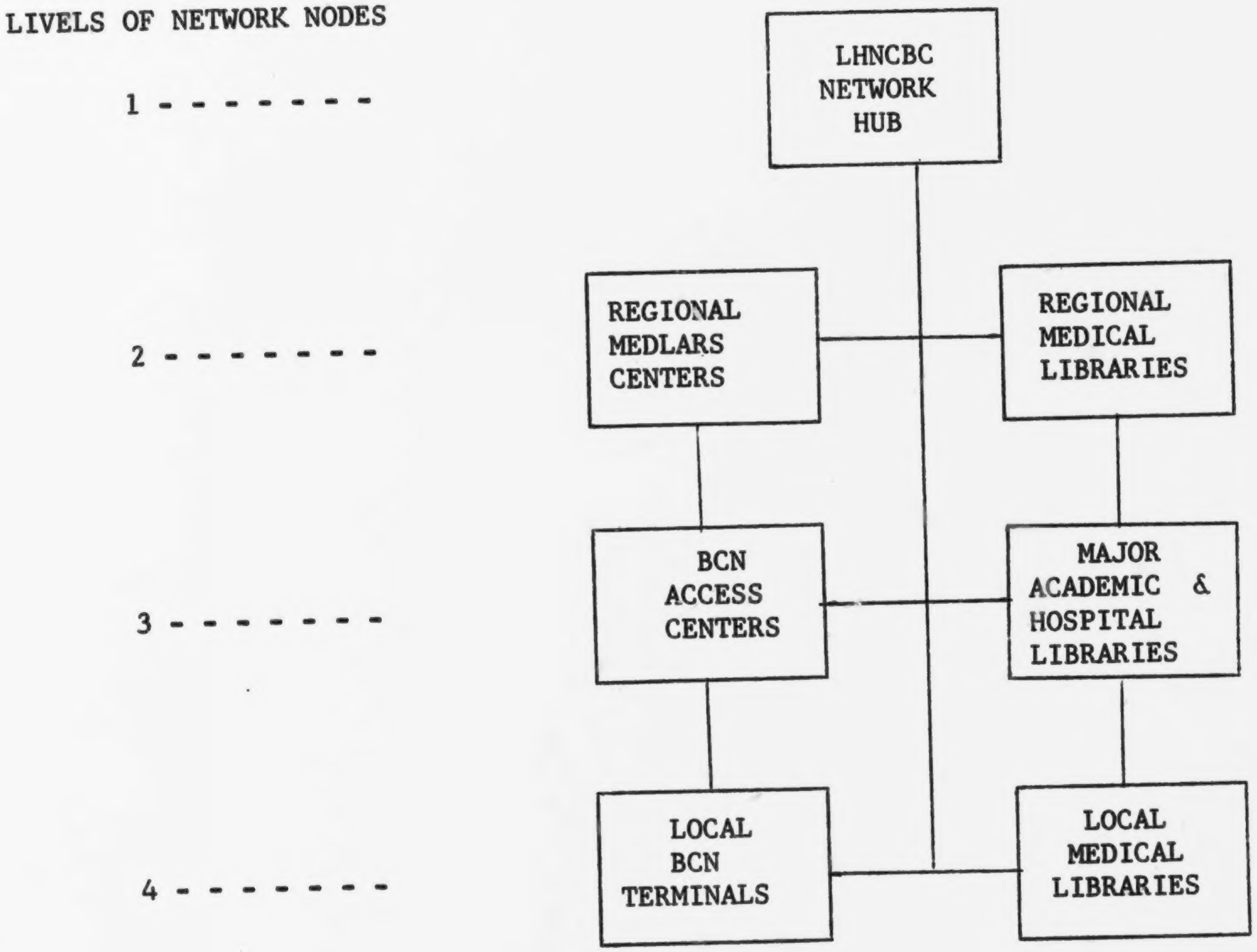
Level 2 - MEDLARS Centers/Regional Medical Libraries
- Assist in input processing
- Major computer and communications facilities
- Data bases accessible on-line
- CBC files accessible through terminals
- Horizontal communications links with other MEDLARS Centers

Level 3 - Regional BCN Access Centers
- Terminal access centers with I/O Services
- Linked to the CBC and MEDLARS Centers for On-line access and message transmission
- Linked horizontally for message transmission

Level 4 - Local BCN Terminals
- I/O devices for message transmission to all levels
- Not on-line to computer files

THE LISTER HILL NATIONAL CENTER FOR BIOMEDICAL COMMUNICATIONS

LIBRARY COMPONENT OF THE BCN


LIVELS OF NETWORK NODES
1
2
3
4
LHNCBC NETWORK HUB
REGIONAL MEDLARS CENTERS
REGIONAL MEDICAL LIBRARIES
BCN ACCESS CENTERS
MAJOR ACADEMIC & HOSPITAL LIBRARIES
LOCAL BCN TERMINALS
LOCAL MEDICAL LIBRARIES

RELATIONSHIPS WITH OTHER ACTIVITIES

- Identifiable "communities" of activity
    - Education
    - Health Services
    - Library Services
    - Communications
- Specific related activities
    - Regional Medical Program
    - Medical Library Assistance Act of 1965
    - Library programs within Office of Education (ERIC)
    - EDUCOM
    - SUNY
    - National Libraries' Task Force

ELEMENTS OF SYSTEM DESIGN

- Action Environment
  - Where
  - When
  - With What
  - With Whom
- Design and Description of Action
- Target Environment
  - For Whom
  - Where

SCIENTIFIC METHOD OF PROBLEM SOLUTION

- Recognize Indeterminate Situation
- State Problem in Specific Terms
- Formulate Working Hypothesis
- Devise Controlled Method of Investigation
- Gather and Record Data
- Transform Data into Meaningful Statement
- Arrive at Assertion
- Relate to Body of Established Knowledge

THE BCN MANAGEMENT PROCESS

Purposes

- To delineate
  Requirements, Policies, Procedures
  for the
  Conceptual, Definition, Design
  Development, Acquisition, Initial
  Operational
  phases of the program

- To prescribe
  Significant management actions
  for
  Integrating and fulfilling responsibilities of organizational elements
  involved

BCN DOCUMENTATION SET

- Statement of Requirements:

  Presents basic philosophy and concepts for BCN, as dictated by the total needs, into a statement of requirements

- Technical Development Plan:

  Translates the statement of requirements into a coherent description of a network which satisfies user requirements

- Network Engineering Plan:

  Refines the defined system requirements and translates them into design requirements leading to system specifications

- Network Management Plan:

  Formalizes a structure of management efforts to establish and maintain positive management control of the progress of the development of BCN

STATEMENT OF REQUIREMENTS

General Requirements

- Needs of society for improved transfer of Biomedical skills, knowledge, and information
- Basic concepts and philosophy for BCN

Specific Operational Requirements

- Delineate and/or define:
    - Services and products of BCN to meet needs
    - Functions and operations required
    - User characteristics
- Define services and products in terms of:
    - Quality
    - Quantity
    - Timeliness
    - Reliability
    - Accessibility
    - Format

TECHNICAL DEVELOPMENT PLAN

Defines operating environment

Prescribes general parameters of network

Identifies resources

Outline of content

. Concept	- General description of operations - Recapitulation of requirements
. Components	- Network organization - Major characteristics - Operating parameters
. Network Integration	- Engineering description of components and communications - Related networks
. Users	- Refinement of user characteristics - Impact on network re .. Location and type of facilities .. Information to be communicated
. Resources	- Estimates by component and by FY . Procurement . Construction . Contracts . Equipment rent . Communications . Salaries

NETWORK ENGINEERING PLAN

Provides

- Definitions of systems of BCN
- Refinement of systems requirements
- Individual system designs

Objective

- Make possible selection of best design approach
- Ensure that system is designed at least possible cost

THE BCN MANAGEMENT PROCESS

Objectives

- Ensure effective managment through network cycle
- Balance factors of performance, time, cost and other resources to obtain BCN
- Minimize technical, economic, and schedule risks
- Control changes to requirements
- Provide documentation supporting decisions made and actions taken
- Establish a blueprint for staff guidance
- Manage and control contractor efforts
- Identify and schedule significant actions and effect accomplishment
- Establish requirements for flow of information between managers and organizational elements
- Undertake necessary R&D for BCN

Discussion

Rogers: Any questions or comments?

Shoffner: How will this relate to other centers such as the Parkinson Center? Has there been any thought about that?

Simmons: There very definitely has. I mentioned that one of the elements of the network was one of the components - we call a specialized information service a component - and if you remember one of the constituent parts of that component will be the information analysis centers. When we talk about the specialized information services component as distinct from the library services, the difference is that we're providing information, answers to questions, and the data itself, rather than documents or references to documents. Actually, the Parkinson Center can be known as an information analysis center under the National Institute of Neurological Diseases and Blindness. So it definitely relates. Exactly how we tie in, what role we play, and what access we have to that system, these are some of the immediate problems before us and what we're trying to develop. There is an Associate Director of the Library who has the responsibility for specialized information aside from the research and development to build such a network. His principal area of activity is in just this area. I realize this is identifying the problem, not the answer.

Hammer: Is there a time schedule for the development and implementation of this network?

Simmons: No, there's no hard time schedule but there's a real urgency we believe. The critical item, of course, in any progress here will be resources. We are currently awaiting word about what the creation of the new Lister Hill Center is going to mean to us at the Library in terms of resources. Our research and development activity at the Library has been eliminated as a line item in the budget, and replaced with the Lister Hill Center. We have the framework of a technical development plan that we had created - violating some of the principles I set here because we jumped into the middle - to try to demonstrate what we wanted to do. And it has a resource schedule in it. It calls for FY 70 to have about two and a half to three million dollars available. We're not at all sure where we will stand with those resources. If we get the resources we've asked for in the TDP (and we're almost assuredly not going to get those kind of resources) we would have had a program that would carry us over five years, which should see us to some kind of a reasonable completion, at least on major elements of the plan we're talking about.

Kilgour: Do you look upon this network as being an overlay on a larger network, or as being completely independent of another general national library network, or haven't you thought quite that sharply about it?

Simmons: There are lots of plans in this area and talk about the development of a formal national library network. We would look upon ours as a component, at least in the library services area or as a participant within that framework. There will be interlocking, obviously, in a matrix arrangement. Part of what we're talking about is directly associated with medical education, as opposed to what might be called traditional library services. That part would fall outside of a general library network probably, but the whole effort would be a part of that general library network plan. This would be our libraries services component, so it would end up being parts of two families really. This is in part the goal of the National Libraries' Task Force, for example. Here we're a specialized library trying to establish a subject network, but still relate in some reasonable fashion to the rest of the library family in appropriate areas.

Reimers: This throws a tremendous burden on the library now at the end that should be the recipient, it no longer is at the end of the chain. The librarian becomes the node and each library becomes the nexus of a library network. Each individual library will be the center of the whole network of libraries services that will lead up to it; isn't this going to throw a tremendous burden on the poor, overworked librarian to decide which network he has to address?

Simmons: Obviously there are going to be some very difficult and complicated interlocking activities, but in fact we're trying to accomplish exactly the reverse. One of the principal organizations we're trying to provide assistance to is the hospital library. Let's look at the fourth level of our component, the local hospital library. For example, the library in your county hospital in the state system is probably administered by an individual that's had no formal library training. We're trying to enhance the resources available at that library through a networking concept for the kinds of reasons that I tried to demonstrate here. One of them is continuing medical education. One of the principal points is that a general practitioner or doctor in the local county has contact with his peers, in that hospital environment. What we're trying to do is to give that hospital environment the kinds of facilities that would bridge the gap between medical research at the National Institutes of Health and the local practitioner - we want to get some of the data out where it can be used to affect medical practice. So, we're trying to do just the opposite of what you described. Instead of placing an extra burden, we're trying to provide assistance and resources that they haven't had before. Now, at the level two installations, which would be major medical school libraries, for example, they are going to have to accept some responsibility to the smaller institutions in their geographical regions, if we're going to reach these people. We can't do it all from Washington and I don't think anybody really wants to. We're working on a decentralized principal; that's why I emphasize management and control. We've been misunderstood, because it's not control on a day to day operational sense, or control of a person's local resources, or on his destiny; it's trying to link him with additional resources to provide a better means to do the same job and to achieve economies based upon participation. I know that this sounds like "motherhood" in a way but we really are excited about this program and feel that it in fact answers some of the problems that we've been unable to lick as individual entities.

Reimers: Aren't we in fact creating a problem for the man at the end? In trying to help him, I think we're also creating some tremendous problems. In addition, we're creating problems for the man in the middle. The end man is not going to have any responsibility beyond his local environment and his hospital environment. It's the second and third level nodes - the people in between - that are going to be responsible for activities beyond their own institutions, and we'll have to take this an institution and a location at a time. I mentioned looking at candidates for our initial phase; this will have to be taken in consideration -- their existing capabilities, the size of their staff, and the size of their holdings. In some cases Federal money has been spent in support of regional medical libraries and MEDLARS Centers. Texas is doing a MEDLARS center service without Federal money, simply by providing them the resources of the machineable tape. I think we're going to have to work on a case by case basis in terms of the initial development to see what the appropriate relationship between the local resources and the local management is and what can be provided from a centralized base.

Adkinson: Following up on Paul Reimer's statement, are you considering information from other than biomedical sources?

Are you going to pull in the chemistry, pharmaceutical, etc., and repackage them so that the fellow at the end gets everything and doesn't have to worry about it?

Simmons: That sounds lkke a nice goal. The answer is, Yes, we're definitely considering it. It's a problem to know what will be the relationships with Biological Abstracts, Biosis, and Chem Abstracts. We already receive data from Chem Abstracts in machineable format, to go into an auxiliary chemical module in our MEDLARS system, so that there can be a link between chemical names and structures to get into the data that's in the literature. We recognize some of these factors as problems and don't have the answers. Of course, this brings up the whole relationship between the Federal activity and private enterprise, or a society's activities. The fact is they're charging for services and we're giving ours free. I don't want to belittle any of these problems, and I don't want to act as if we have a panacea now, and all we have to do is push the button and we've got the system going. We recognize all of these problems, but we believe that they're not insurmountable. For example, Dr. Bergstrom, Head of the Karolinska Institute in Sweden, just returned from negotiations with Biosis. I think there's going to be an exchange of machineable data in the coming year on a free basis which will begin to explore the kind of relationship that should exist there. Each one of these problems is being studied under its respective part of our program.

Kilgour: I'd like to pursue this line of questioning further. I think Paul Reimers was interested in what the local requestor is going to do to get information that he wants for a medical application; let's say he's a physician or a resident in a hospital; what he wants is something that comes from sociology that would not normally be in the BCN network as such. Is he going to have to decide which network to put his question to? The question I think is, do you envisage a network where the user is going to have to make the original diagnosis as to which network he's going to enter, or is he going to make one entry and then automatically referred unbeknownst to him to get the right answer?

Simmons: I'd like to say that latter is going to occur eventually. I think realistically, right now, the person primarily in the medical field, when he has a question outside of his field will have to consult a reference librarian. The reference librarian will know what resources are available, and where to go to get an answer. We're not going to take any quantum jump from our present situation where we're going through that kind of a link.

Kilgour: You would provide that solution rather than place the burden on the questioner?

Simmons: Absolutely. Eventually, we'd like to have the private physician have a console in his office where he could sit down and ask his question and have the network automatically get his answer. We have a long ways to go before this will be realized. The plan we're talking about, in the five years we're talking about, does not include this type of service.

The University of Chicago's
Book Processing System

Charles Payne
System Development Librarian

The University of Chicago Library

A Paper Prepared for the Stanford Conference on Collaborative Library System Development, October 4-5, 1968

The three project summaries given earlier outlined various individual approaches to the long-range problems of library automation. This paper will go into more detail about the Chicago approach: why we have proceeded the way we have; our results so far; and, also, what we have yet to do to complete our first phase development. The Library of the University of Chicago is now into the third year of its project to mechanize bibliographic data processing. This project has been funded, in part, by the National Science Foundation.

The current project staff is as follows: the Library systems staff consists of 3 full-time persons, in addition to myself, and, of these, one works full-time on operational and cost studies and one started to work this week; the computer system and programming staff is approximately 2.5 F.T.E., which is down from a high of 5; the data input clerical staff varies from 5 to 6 F.T.E.

One of the goals of project development has been to eliminate or decrease much of the manual record generation, processing and maintenance normally associated with the library technical processing operations --acquisition of materials, fund accounting, payment processing, cataloging, book preparation or finishing (binding and labeling), book distribution, and catalog maintenance--and to reduce this manual paper work by use of computerized data processing. We have felt from the beginning that the handling of bibliographic data was the key factor in library automation. Following logically from this we have worked from the beginning with the concept of the unit record--integrating all of the various processing, bibliographic, and operational data within a single record in the machine file. Our design was to be able to create a record and to update it at any point in the technical processing operations; to enter data, partial or complete, at any time and subsequently be able to use, amend, or correct these data; to signal desired output at any time and get it at the desired time in the proper format, and positioned in an array designed for easiest use. This is what we now have in operation.

This integrated design has been, most certainly, more difficult in the execution than a more standard functionally oriented design would have been. It required initial development of plans including all phases of technical processing; it required a very large effort to define bibliographic and other data elements (at a time before MARC definitions were available); it required, in our earlier phases working with computer operating software that was inadequate and unreliable so that substantial effort went into debugging, modifying, and extending the operating software as well as into development of the library system supervisory and utility software and the applications programs. Further, because of the long-range nature of a complete system development, and because of some intense pressures from both within and without the library to become operational, it was decided to build a full scale, operational system--full-size, full-rate--from the start and to implement the various capabilities of the system into library operations as soon as they became available. We took this approach rather than to initially build a test or model-sized system. As many of you know, no complex, interrelated set of programs for on-line operation can be completely debugged quickly. It is a matter of testing all possible sets of conditions. We have, on occasion, experienced gross failures of certain programs some months after they were considered operational when a different, untested set of conditions would occur. (This kind of problem, however, would undoubtedly also occur following any system change-over from a model-sized to a production-sized operation.) Add to this the extreme difficulty we had (before we learned how to do it better) in adding new programs and capabilities to a highly interrelated system without fatal upsets to other previously stable parts of the system. You can probably understand why this development has, at times, sorely tried the patience of almost everyone involved--the programmers and computer system people, who were under pressure; the library staff, who had responsibilities for ongoing operations, whether the computer system was up or down; and even I fear, at times, we tested the patience of the library administration.

We attained eventually (this year) a level of development which begins to make the effort seem worthwhile. We have started to reap some of the benefits of this method of development.

I have recounted some of the hazards and now I want to list some of the positive aspects and long-range benefits:

1. The system as developed is not a purely theoretical one nor a simplistic version of library operating requirements. The operations and products have been tested and developed in use. For example, the catalog card format programs have become extremely versatile and can handle any of the wide variations in bibliographic records that we encounter, within the limitations of character set or as long as the record does not overflow to more than 16 cards.
2. We were forced to work with the whole range of bibliographic and processing data elements from the beginning. In effect we undertook the most complex aspect of development first. This has already paid off in terms of making subsequent development easier.
3. In spite of the fluctuating consistency of operations for an extended period, the system did during this time produce large quantities of usable and useful products to augment ongoing manual operations.
4. Perhaps the most important benefit is that the programs, because of necessary changes, have been honed and sharpened and standardized in ways that make operation more efficient and that make any changes and additions much easier to accomplish.
5. The system as developed is beautiful, in a sense, in its independence of the terminal or printing equipment used for input or output. We have no immediate plans to make use of CRT terminals, but if this utility were to be incorporated, it would be simple to do, in terms of programming changes. It is also relatively independent of how we want to operate--in an on-line or in an off-line, batch mode. We were able (and I could even say forced) to re-evaluate our original ideas concerning on-line operations. This allowed us to utilize on-line operation where it was most beneficial and to go to batch operation where that mode proved to be more efficient. To explain what I mean, I need to mention our experiences with errors and error correction. As it turns out, error correction quickly becomes the key, critical factor in machine processing

of bibliographic data. Error rates are atrocious. The use of clerical typists to keyboard data in all the many languages of the world results in high error rates and there is not much that can be done about it. We have, therefore, tried to make error correction as easy as possible. We have at least three levels of error correction. The first level makes the corrections before being read into the machine system. Not counting first level errors, the rate runs to about 25% of the item records processed. We have found that on-line operation is most essential for data read-in (not keyboarding--we keyboard into paper tape) and logical error message response, essential for calling up and receiving item record printouts (this when things are so mangled that the hard-copy worksheet from keyboarding does not help), and also for the error correction read-in and its error message responses. In our system this allows us to make error correction at any time right up to the minute that the catalog cards or other products are produced. Many items go through the error correction cycle more than once. A very high percentage of our 25% error items are corrected in this manner so that their products are produced in the batch with which they started. On the other hand, we did not find that on-line control of routine output production was very useful and we have abandoned it to a large extent. If a change occurs making printout equipment in the library practical, we can resume on-line control very easily.

6. The final benefit that I want to mention is that the system as evolved is ideally suited to the use of externally generated machine-readable data, such as MARC II data, and we intend to take advantage of this as quickly as possible.

We still have a lot of work to do on our system, with a number of applications to incorporate, before we will have completed our basic Phase I development. This Phase I development is our major effort for the first three years of the project. I would like to describe what we have, what we are working on, and what we are planning. Even though this system is called the Book Processing System, that name is not totally accurate. The system is designed to cover not just books or monograph processing, but also serial ordering, fund accounting, payment processing, and cataloging--most of serial processing except for serial issue check-in and serial holdings records.

Data input is, of course, a prerequisite to any machine data processing and this was one of our earlier accomplishments, We initially developed our own tele-processing software and have maintained it, for reasons of efficiency, even though utility software has been available to us for some time. The Library has developed a Data Processing Unit which handles input and output on the machine system. Work is channeled from library operational departments to this group. We do not have input equipment in other locations. Also, keyboarding of data is not directly on-line, but into paper tape. The paper tape is utilized, in effect, as a giant buffer. First level error correction can be added to the tape anytime subsequent to the error or on a second tape to be read after the first. In the latter case, the machine receives both the error and the correction and processes to make the correction.

All data is input in the form of tagged data elements. One type of data element merely contains information to be maintained. A second type of data element not only contains data but initiates action within the machine system depending upon what the data are. A third type can initiate action merely by being present with no regard as to the data content. Data goes in as a string of tagged (and thus defined) information and they are maintained in the machine file in this way. Formatting is strictly an output processing function.

Signals for output are also input as tagged data elements and result in the required ouput array, or stack, building. On a signal to print, either on-line or as a batch processing job, each item in the stack is sequentially formatted and printed out. Catalog cards are printed in arrays for the desired catalogs or other locations receiving cards; the arrays are in filing order whether for main entry catalogs, author-title-subject dictionary catalogs, or shelflist catalogs. These programs are all operational; they are quite sophisticated and extremely versatile. No further changes are currently planned in this area. Catalog card production has been the most affected of our operations by the systems changes we have gone through and this production has suffered considerably through up and down cycles of production. I decline to state that we are finally doing 100% of the Roman alphabet cataloging on the system, although we have made the push this week to go to 100%. We have recently been processing about 150 cataloged titles per day, producing an average of 11 cards per title, for a total card production of about 1500 to 1700 per day. This rate represents 75 to 80% of the current Roman alphabet cataloging work.

We utilize the same bibliographic data, with different formatting and a different output array directory, for book cards and pocket label production. Catalog card sets are based on title, but book finishing products are required for each physical volume. The programs handle this by an expansion of a relatively simple holdings statement. It is not unusual for this operation to produce 20 or 30 sets of cards and labels for multiple volume and multiple copy materials.

This production operation also covers a wider range of materials than does catalog card production. Cards and labels are being produced for virtually all materials in Roman and non-Roman alphabets. The Romanized, or transliterated, entries and titles are used. This provides us with a machine record that is acceptable for some uses though not for catalog cards.

The system has handled book card and pocket label production for the library for a long time, though not always with the one-day currency desired.

The programs for computer formatting and printing of purchase orders have been completed and tested except for the final full production-run testing. We are planning a coordinated effort to get this implemented into library operations as soon as we all get back to work. Programs are also completed for production of a daily fund commitment list. This would be the first step of a more complete fund accounting system. As this list makes use of order data, its implementation is dependent on that of the order printing operation.

We have proceeded with order printing development even though we are working with the CLSD group in a joint design effort covering all of acquisitions work. There is no great conflict here, however. Any emerging joint design that Chicago could adopt would need to be hung on our existing data processing system. Order printing requires a set of data element definitions and some forms and formats. The set of data elements we use are not in gross conflict with the CLSD list and could be easily modified. Forms and formats will probably, of necessity, be governed by local considerations for some time in any case. We look forward to CLSD efforts that would go beyond purchase order generation, perhaps to include telecommunication with large vendors.

The areas of CLSD effort that are of most immediate interest to the Chicago development are fund accounting, payment processing, and management reporting.

We have substantially altered our thinking, particularly on payment processing, since these discussions have begun. The daily fund commitment list, mentioned earlier, is really an interim, partial fund accounting effort. It is likely that further work in these areas will await results of the joint design effort.

Chicago also has an automatic overdue order claiming operation designed and ready for programming. We will not proceed with this immediately, pending further discussion with the CLSD group. To date, CLSD discussions of claiming have not gone far enough to resolve conflicting ideas, although we all agree to the need and, in some ways, the method of application.

We are both planning for and working toward prompt utilization of the MARC II data in our system. Our programming staff has studied the MARC II format, as released, and have developed plans for conversion of MARC data to meet the requirements of our systems. Program coding will not begin until further and final information is available concerning the MARC II format. We are also developing our plans so that MARC data can be incorporated into our system in the most efficient and utilitarian ways. We have decided that, initially, we will attempt only to convert MARC data to the Chicago format. We will not attempt two-way communication initially simply because our cataloging is not in sufficient depth to meet the MARC requirements and because there has been no clear indication from the MARC staff as to how they intend to cope with this. We have a system well suited to the use of MARC data. Operationally, we intend to process the MARC tapes into the system as they arrive and make use of the bibliographic data elements as early as possible in our processing--even for ordering, if MARC is fast enough.

We, too, are convinced of the necessity of nationally generated bibliographic data. We look forward to the point where MARC data can relieve us of substantial portions of data input and, we fervently hope, error correction.

We have other products and operations in the planning or designing stages, including new book lists to be generated for subject or departmental locations that receive books. These subject book listings would be yet another product use of the bibliographic data used many times before. Because non-Roman alphabet materials have been included for cards and labels, these lists would be comprehensive if not elegant.

Another area of control that we are very interested in is bindery shipment control, with bindery tickets and finished book distribution lists as further products of the system. This development has not proceeded to the programming stage yet and will probably be one of the last efforts of the Phase I development. It has at least one interesting application for catalog card production, book card and label production, and new book listing and this is the timing factor. One may not want to advertise new books or prepare products for their finishing while the books are still at the bindery.

This system, as described, both the completed and the uncompleted applications comprise the basic Phase I design. We plan to have much of the system in operation by the end of the year. It was intended to stabilize the routine operations of the library and to provide a sound, modern operating base from which to build in the future to the more sophisticated, information-access libraries we all hope for.

(Discussion follows next paper.)

3

The University of Chicago's
Book Processing System:
Programming Development

Mrs. Kennie Hecht
Programmer

The University of Chicago Library

A Paper Prepared for the Stanford Conference on Collaborative Library Systems Development, October 4-5, 1968

The system with which the Chicago Library Automation project is presently running consists of an IBM 360, Model 50 computer with 512 thousand bytes of core storage, 10 2311 disk drives, a 1403 Model 2 high-speed printer, 2 9-track tape drives and 2 7-track tape drives.

The software system is IBM-OS-MVT. That is multiprogramming with a variable number of tasks. This means that a variable number of programs can be running in the computer at the same time. The library tele-processing programs are in the computer from 6 to 14 hours a day while other programs are being run and other tele-processing operations may also be going on.

We have two print trains for the 1403 printer. A standard one which has only upper case letters, numbers, and very few special characters, and the special library print train which has upper and lower case letters, numbers, and many more special characters. The standard train is kept on most of the time because of the greater speed it permits. The library train is mounted whenever library printing production is being done. Almost all of the regular library production printing is presently being done by regular computer operators on the midnight shift.

Eighty-five operational computer programs have been developed thus far. They vary in size from 96 bytes to over 4200 bytes. These programs are stored in two libraries on the 2311 disks. One of these libraries on the disk is a library of programs in which individual programs are stored. The other is a library of phases (a phase is a group of programs linked together and operating as one). In this all the programs for a single function are stored linked together under one name. When the on-line tele-processing receives a command from the remote terminal to do input--it calls in from this library the input phase as a single package. When it is commanded to print a record, it brings into the computer from the disk library the file printing phase, etc.

Those processes that are batched jobs work much the same way. A small deck of cards (usually under a dozen) is read into the computer from the card reader. This small deck brings into the computer from the disk library the phase it needs, and initiates processing.

I would like to give you a brief description of the library system from the programming point of view. The system as seen from this end can be broken down into a number of phases on the basis of function. The phases are as follows:

1. The tele-processing phase: This phase consists of 13 programs which control the passing of data back and forth between the computer and the remote terminals at the library.
2. The command processing phase: These 2 programs accept the commands from the remote terminals at the library and initiate the appropriate action, bringing in from disk storage whatever processing programs are required.
3. The input processing phase: This phase consists of 16 programs These programs check each incoming record against the library computer file to see if it is a new record or the updating of an existing record; scan the input for invalid data tags; edit out unwanted blanks and control characters. They scan for output requests; create an entry in a list for those records with output requests; perform the necessary changes in the record depending on whether the new data is an addition to the record, a correction, a deletion, or a totally new record; and write the new or updated record on the library's computer file.
4. The utility programs: Two of these programs print out records from the file as they appear in the file (this may be done on the remote terminals or on the high-speed printer); a second set of 5 programs reorganize the file, check for file errors, and provide a backup copy of the data file.
5. The distribution interpretation programs: This consists of two phases. The first phase of 12 programs takes the list of records requiring output, selects those which are for catalog cards, reads the record, checks it for errors, creates a card by card list and sorts this list by location and entry. It saves the list on a disk file, prints the list, prints any error messages, and prints a count of cards by location. The second phase of 10 programs takes the list of records requiring output, selects those for orders

reads the record, checks it for errors, creates an order by order list, sorts it by dealer, saves the list on a disk file, and prints the list, any error messages, and a count of the orders by dealer.

6. The catalog card printing phase: These 18 programs select an entry from the expanded list of catalog cards to be printed, read the selected record, do a quick check of the record for mandatory data and redundancies, format the call number, format the lines format the cards (main entry, added entry, shelflist--single or multiple, as needed), and either output the formatted card to some printing device or print a message about errors found in the data.
7. The book card and pocket label printing phase: These 19 programs select an entry for cards or labels from the list of records requiring output, read the selected record from the library's computer file, do a quick check of the record for mandatory data and redundancies, format the call number, format the text, and print out the formatted item or error message on some output device.
8. The order printing phase: These 16 programs select an entry from the list of orders to be printed, read that record in from the library's data file, do a quick check for mandatory data and redundancies, format the text of the order, sum up the fund commitments by fund, and output the formatted order on some printing device.

Looking over these phases again, you will see that no phase consists of a single program. The reasons for writing the phases as sets of small programs, rather than having each phase as one large program, are the result of some planning and much experience. These reasons are 1) to write as many re-usable routines and programs as possible, 2) to make the phases as device independent as possible, and 3) to make the phases as easily maintained and changed as possible.

Point one: To write as many re-usable programs as possible. When I was describing the phases, you may have noticed that many of them included the same function. For instance, input processing, the utility programs, the distribution interpretation, the catalog card printing, the card and label printing, and the order printing phases all must read records from the library's data file into the computer. So one program is written that reads the record from the data file into the computer. It is written for maximum efficiency and minimum running time cost. All of these phases can then use this one program avoiding any duplication of programming effort and cost. Perhaps the most telling example of the savings in time and effort that are possible, is the difference between the programming necessary to implement catalog card production (the first formatted output phase written) and that necessary to implement order printing (the last formatted output phase written). Implementing the catalog card printing required the writing of 18 programs, an effort that took many, many months. However, these routines were programmed to be usable in more than one phase. So when it came time to implement order printing, a set of 16 programs, only four new programs had to be written. The other twelve were taken exactly as they were from the catalog card phase. The fact that we had to write four new programs illustrates the fact that we have not totally mastered the art. Ideally, we should have a generalized output formatting phase that requires the changing of only one program from one type of output to the next.

Point two: To make all the phases as device independent as possible.

This is one of the lessons we learned the hard way. Computer hardware and soft-ware is a rapidly changing field. It is desirable to be able to take advantage of new equipment and system advances as they become available. To do this we have isolated those parts of each phase that require device dependent coding. Then when the equipment changes, only the isolated program need be changed and the basic operation of the phase is not touched. Thus, if it should become advisable for the library to change its input method, or to make it more flexible, only one program need be changed. The library could start inputting directly from the keyboard of a remote terminal in addition to the present paper tape, with only about a week's programming effort. With the inclusion of a conversion program they could input data into their data files from outside sources such as MARC II. This flexibility is necessary in order to have an enduring automated system.

Point three: To make the phases as easily maintained and changed as possible.

The small size of the individual programs is the single greatest aid to maintenance possible. It is by nature easier to understand a small isolated program than a large complex program. Then when there is an error in a phase the programmer can separate the individual program containing the error and work only with it to correct the error. This may mean the difference between having to keep in mind 40 or 50 pages of coding and having to understand a three page program.

The library system is a developing system. The library's file organization and data record structure permit the addition of new data elements to the data base. To avoid having to change the programs whenever additional data elements are added to the data base, we use tables. The input phase has tables of valid data tags, the output programs have tables of the data elements to be included in the particular output, the sorting programs have tables of articles to be removed for proper sorting, to name a few. If the library wanted to add a new data element to its records, for example, a national book number, and it wanted this number printed on all catalog cards, the designated tag for the book number would need to be added to the table of valid tagging codes in the input phase program and the tag would need to be added to the table of data elements included in the catalog card output. No other programming changes would have to be made.

Needless to say, we have not always succeeded in carrying out these three points in the programming. But the effort to do so is beginning to pay off in the decreasing time and cost needed to implement each succeeding phase of operations.

Discussion of papers by Payne and Hecht.

Weisbrod: You mentioned fairly early in your talk that you had learned that on line control of production functions was not particularly useful. Can you explain that statement?

Payne: We initially started operations using IBM 1050 terminals in the library for both input and output work. At that state we didn't have a high speed printer available to the project. We were printing catalog cards at a relatively slow rate on 1050 terminals. The control of this was all on line from the library end. We dialed in, gave the proper language commands to start, assigned the device and all and started the printing. It was slow on the 1050, but it was still a large batch, but there was no particular value in the library's being able to say we want to start it now or stop it now. Batch processing of the catalog card printing overnight and receiving them the next morning works every bit as well.

Weisbrod: In other words what you're saying is that it was essentially batch processing anyhow and the ability to initiate batch remotely wasn't particularly thrilling?

Payne: That's right. Maybe kind of fun, but not very useful. Some day, it's quite possible that we would want to have an output printer in the library with remote connection to the computer. Then we would indeed want to be able to schedule the batch run, simply so that we would know when to put the proper forms on. But, in terms of the daily processing operations of our system for the library, this remote control is not very good.

Boice: What made you change your input from 1050's to paper tape?

Payne: I've always been on the same system. We've used 1050's from the beginning and we've used paper tape from the beginning. We type using a 1050 keyboard with a 1050 paper tape punch, we make a paper tape, and then use this 1050 paper tape reader to read it over the line.

Shoffner: Would you care to discuss any of your unit costs?

Payne: Did I lose that sentence? I did have a sentence somewhere at one time that said something to the effect that as we implement more of these operations and get into a more sustained production operation through the rest of this year, we will begin to generate cost-effectiveness data, which will then tell us probably the whole story.

King: Is this IBM 360/50 used exclusively for library projects?

Payne: No.

King: Is it a computer center facility?

Payne: Yes, we have some priority in permanent core allotment, but it is a production computer.

King: Now is this upper and lower case chain the standard TN chain, or is it something you designed yourself?

Payne: It was the standard TN train, and we made a fair number of changes. On one of the sets we substituted diacritical marks and other special symbols in place of the superscripts and some other seldom used graphic symbols, so that we have a full TN train, but we have also twenty or so additional symbols.

Fussler: May I ask for equal time? If Burt Adkinson is still here, I want to be careful to speak to the staffing problem, so that there isn't a mutual misunderstanding in the room when we leave.

The figures I quoted were from your report, Payne. Perhaps misinterpreted. I think where I came out was that we had, and I just am going to repeat what I said, that these were totals of staff used in the last fiscal year. They're not representative of necessarily current staffing level, and the FTE total of library systems programming staff came to approximately ten people which I thought it was an extremely small staff in terms of the accomplishment. The second point I'd like to make in rebuttal [Laughter] to the speaker's remarks that you just heard has to do with the percentage of card catalog production, as to whether it's 100% of Roman alphabet or not this week. I was given this information on very high authority by the head of our Systems Development! The third point I would like to make is that we have been through a series of computers and a series of computer operating systems that have had a detrimental effect upon application programming and the work of the staff. I'm led to believe in conversations in corridors at various meetings that this is not an uncommon experience, and that the advertised upward compatibility of all systems and all hardware is still to be realized, but it has proved a substantial drain on staff effort and, to some degree, systems staff morale, as I think you can all imagine. The progress that's been made in the face of these frustrations which it seems to me Charles Payne and Kennie Hecht have displayed remarkable restraint in not mentioning, has been significant. The fourth and last point I would like to make again, is to emphasize what Charles and Mrs. Hecht have said; the process developed thus far creates a very powerful data base in terms of library operations and library processes, with a wide variety of potential application purposes with, it is hoped, relatively small further investments in the programming effort to utilize this kind of product, and it makes in consequence, we think, the emerging potential services of a library, both in relation to technology, and in relation to needs for data access, an important aspect of the system.

Welsh: Mr. Rogers I'd like to cross-examine the last witness. [Laughter]. I was formulating two questions, and maybe if I expressed them, maybe then you can clarify the doubt that remains in my mind after the distinguished librarian from the University of Chicago has spoken. Did you say you were inputting at the rate of 175 titles a day?

Fussler: 150 titles a day.

Welsh: What's the size of the data base you have now accumulated?

Payne: I haven't the foggiest notion. We've been through every conceivable rate of daily operation, between 0 and 150 at one time or another.

Welsh: That's the point of this question. Join the club! My next question was, how many people, how many full time equivalents do you have inputting at 150 records per day?

Payne: Well, in terms of simply inputting, it can't be more than two. We only have two machines available and there is not somebody sitting at them all the time, and we operate only on a standard day.

Welsh: Are they punching from coded work sheets?

Payne: No, they're working from almost anything. Anything from an LC card with beautiful data on it, to a record that may be part photographed, part hand-written in middle European script, or part typed on it, we use a variety of different kinds of pieces of paper that tend to arrive by different routes. There is no standard.

Reimers: Are your key punchers editing, interpreting and adding tags?

Payne: Yes.

Burgess: I'm still trying to get at the hourly rate. What's the hourly input rate?

Payne: I don't really know.

Burgess: The gal we had when we were designing our system was timed to compute loading factors. She ran in about 40 in an hour on these records, and she was editing the tags, too.

Payne: Complete bibliographic records? I think that's fairly high.

Unidentified Voices: That's very, very high. That's about 60 seconds per record; I'd give her a job anytime. [Laughter].

Burgess: Our operational experience is nowhere near this.

Payne: Well, the maximum 1050 character rate is only 14.8 characters a second. I think she may have exceeded that. [Laughter].

Thomson: Am I correct in assuming that once the catalog card is produced, the data is removed from your disc file?

Payne: It is removed to a different file. It isn't removed; it's taken out of the active processing file and put in an historical file.

Unidentified Voice: How Many 2311 disc packs does your current in process file fill?

Payne: It's on two now.

Hecht: Time data is on one and indices and several little files on another one. There are two discs on line all the time the library is running.

King: Are the programs on those two disc packs also?

Hecht: Those areincluded in the files on that device.

Weisbrod: When did your card production begin?

Payne: I no longer remember.

Fussler: We first got useful cards on the 1050's at the end of 1966-67 fiscal year; then we began switching to DOS. The DOS high speed printer cards began to be available May of 1968. Current production on the IBM 360/50 is indeterminate, but fairly large volume began in August/September of this year.

Shoffner: How many tape files have you got now with bibliographic records?

Hecht: Are you trying to determine the approximate number of bibliographic data records we have?

Shoffner: Right.

Hecht: An estimate would be that the active on line processing file probably has 15 to 20 thousand data records. The historical file, which has not been counted for many months, now probably has maybe 50 or 70 thousand data records.

Shoffner: How much program have you written, either in terms of source code lines or in terms of object space on the disc or something of this sort?

Hecht: I really don't know offhand. Approximately one card file full of programs, plus two old systems changes; we probably have about two more files filled with programs that were outdated by system and equipment changes.

Shoffner: This is all assembly language coding isn't it?

Hecht: Yes.

King: How big is your teleprocessing partition?

Hecht: I believe it's about 30K.

King: How about your biggest phase?

Hecht: I would guess maybe 50K. Most of them are still fairly small because most of the previous machines we ran on didn't allow for much more than that in the core space. They were written accordingly.

Boice: Your statements on the difficulties you've had with hardware-software compatability and upward graduation leads me to ask this question. You say you have ten 2311's on the 50.

Technologically, it would seem more suitable to have, say one 2314. You'd be spending less money and have about twice as much storage. Are there any problems you've run into in your hardware-software usage that detects this, or have they just not been changed by the university?

Payne: The 360/50 set-up with the 2311's is quite recent. Four of the 2311's came from our project and other projects on a previous machine. The ultimate equipment configuration here hasn't really been decided.

Weisbrod: This question is prompted both by your remarks this afternoon, and what I might consider to be kind of an absence of remarks this morning from the three people who gave the project summaries, Herman, Paul and Allen. Is CLSD involved at this point in the design or implementation of commonly usable programs, or is it a more general exchange of information, but nothing quite as concrete as that?

Payne: I think that we think we are working on joint design specification covering an acquisitions module for an on-line library operation.

Fussler: But not necessarily a common software package? Isn't that right?

Payne: We haven't reached that stage. We have not done any program development. We're working in the design area now. Common data elements, common forms, common processes, and operations are the areas we are trying to resolve into a joint design.

Fasana: I think the first thing we wanted to accomplish was to see if we could work together.

Logsdon: I know the time is late, but sometimes I'm accused of double talk, but I thought I was very explicit on that point this morning, both in what I said and what I very carefully didn't say. I think it's fair to say that CLSD in its sixth month has had its influence up to now more in the internal exchange of information and influence on our various operations. Somewhere down the road may come some of these things you're talking about. I don't think anyone can say at this point what precisely or how much.

King: This makes the previous question more explicit for me. Is it possible for universities to get these programs? Are they reasonably well documented, so that with some effort some other university could take over your system? What about these bibliographic files, could copies of these be obtained? Is this cooperation going to go that far?

Payne: Actually, we've been handling these kinds of questions by saying, that until the end of the third year, we won't really have things stabilized enough that it would be worthwhile talking to anyone. As far as I know there is no reason why the files couldn't be copied. It's not clear to me that this could be done easily. I think eventually we will want to share our work, but back to our previous discussion about staff. No matter how big it is, it's tiny. We don't relish at this stage spending staff time trying to make use of these things elsewhere.

Unidentified Voice: When is the end of the third year?

Unidentified Voice: In about ten years! [Laughter].

Payne: As Ralph Shoffner says, time is relative. Officially June 30, 1969.

Spaulding: You spoke of error rates and some of them quite large; it seems that at least part of that percentage might be due to the sort of mixed media input. Do you have any feel, or did you say and I missed it, what percentage of errors get all the way into the machine system?

Payne: What results in error messages or actual printed out errors is 25%. If we hadn't cleaned up before, I think it would be uncomfortably close to 100%.

Spaulding: But then you clean up this 25%, or perhaps you mean clean up the 25% of 25%. Do you have any feel for what is finally in the machine when you left it thinking they're right?

Fussler: 25%.

Payne: No, we end up with a small percent, which I can't give you offhand, that is indeed in error when we print out our product. Actually, I don't know what this is. It is made somewhat more cloudy by the fact that there are print out types of errors that fall in there, too. The forms weren't aligned right, or the card cutter for some reason wasn't cutting cards that day, or the guy put them in upside down and ran them through the card cutter. These things happen and this all goes into the ultimate error rate. But I don't know the number of errors that actually go in that are not caught until we actually have printed out a product.

Spaulding: An interesting exercise is to see the error sheets in the NEBHE project that are coming back to the center where they're producing cards, also the kinds of errors that occur on the cards, and difficulty in cleaning them up. Do librarians feed cards back to you in any quantity? Or do they say it was on the computer - it must be right, and somewhere it goes into the catalog? What I really am concerned with is whether automation is causing more or fewer errors than the normal manual rate of error in card catalogs?

Payne: I can only say that I have a feeling. I have a feeling that it is less than the normal manual rate, but that there probably is indeed some small error introduced that goes on into the permanent printed records. Catalogers seem to be rather diligent in finding them and returning them with relish and acerbic little notes. What we don't know is what they didn't find.

Shoffner: Are you keeping any kind of history file on your error correction?

Payne: No.

Shoffner: I have heard indirectly that Dick Johnson kept all of his correction cards from the Stanford Undergraduate Library Catalog Project. Is that correct?

Johnson: When I was at Stanford, we gave a sample of edit lists to Jim Dolby which he used in the preparation of his study. He based his conclusions, I think, on data from Stanford and from Harvard.

Shoffner: The reason for asking the question is that with the massive conversions that are going on or are being planned, it would seem most desirable if these efforts would build into them mechanisms for keeping track of their error correction process, the kinds of errors encountered, so that studies could be made on the kinds of errors that were made, so that we could determine what sort of machine assistance there might be beyond just brute force authority files and this sort of thing. This is basically Jim Dolby's idea to use some of the structure of the English language, or other languages such as the case may be, to try to help out with the machine.

Payne: We could quite easily save the daily error correction tapes, which would tell what had been corrected, that gives you the corrected version, but it wipes out in our system the incorrect version.

Shoffner: So you wouldn't have both?

Payne: Not unless we saved the hard copy work sheet from the original input.

Weisbrod: And the original paper tapes?

Shoffner: That gets pretty large.

Payne: I can tell you just from observation that the very largest share of errors in our input are misspellings of words in foreign languages.

McKirdy: That's one of the things I'd like to ask you about. Who does the editing? And if you are now doing 100% of the Roman alphabet, you must need a number of people skilled in many languages just to catch these misspellings. This is one of the problems we have in the conversion of the shelf list.

Payne: At present, we are using clerical personnel for proof reading who do a character by character comparison.

McKirdy: In other words, they're comparing machine output with various source documents?

Payne: Yes. We hope actually to return some of the proof reading to cataloging. We want to do this particularly where the source document was a pretty crumby piece of paper to begin with, and didn't really look like a cataloging record. We would input and format a record for them to look at in its final form for a final OK. We haven't implemented this, though.

McKirdy: Do you have any idea, any figures on the number of editing hours compared to the number of keyboard hours? How many editors do you need to keep up with your operation?

Payne: I don't really know, but I think it's about one for one; in that general magnitude at least.

Boice: Have you made any attempt, or was any consideration given to preparing a standard worksheet for cataloging?

Payne: We batted this around a number of times, and have not. I don't know whether we ever will.

Economic and Operating Realities of Present Day Hardware and Software in Library Applications

William F. Miller
Professor of Computer Science
and
Associate Provost for Computing
Stanford University
&

Richard N. Bielsker
Programming Manager
SPIRES/BALLOTS
Stanford University

A Paper Prepared for the Stanford Conference on Collaborative Library Systems Development, October 4-5, 1968

Welcome to Stanford. Although I was unable to attend yesterday's session I want you all to know that we are delighted to have you here and we welcome this opportunity to discuss the Library problem with you. This is an area in which I have had, and continue to have a fair amount of interest; I have put some administrative effort into it and, in the early stages, an iota or two of intellectual effort. But I wish to make it clear that we aggressively support the application of computers to bibliographic work.

When Allen asked me to speak he really wanted someone to "tell it as it is." In order to do that, you first have to see it how it is, and then the "telling-seeing" makes for a reasonable aphorism. I admit that I'm not certain whether I see it as it is or not. My colleague Dick Bielsker has sold information systems; I have never done so, although I have sold a lot of other hardware/software systems--perhaps as many as thirty of them over the past fifteen years. From that standpoint I suppose I can venture a pretty good approximation of how it is.

One of the things I have encountered time and again is people asking, "Why is it so difficult to engineer hardware/software systems? What really is the difference between engineering the composite system and just a hardware system alone?" It seems to these people that engineering a hardware system (and we have been doing this for some time) is a lot easier than engineering the composite system. In fact, judging from the reports I've gotten about the informal conversations after yesterday's meeting, I conclude that a lot of difficulties are being encountered in engineering library software systems. Well, if I can talk as "Professor Miller" for a few minutes I will try to explain the real problem here.

The reward, and at the same time the retribution, of software is self-change. That little fact comes back to haunt us and to help us in many different ways. In one of our introductory computer science courses we are told that a program and a machine are essentially the same. That is a very useful idealization, particularly in regard to the dynamics and logic of programming the machine; but in real implementation the self-change part is delegated to the software. Thus one might say: in the real world, engineering of hardware is different from engineering of software or hardware-software composites, because in engineering the latter kinds of systems one deals with machines that change themselves. And therein lies the heart of the problem.

The side effects of self-change are the things that haunt us. They cause all the little and not-so-little bugs that we encounter in interfacing with the operating systems. I'll elaborate on this situation in several different ways.

One of the first considerations in the development of systems is to take a good look at what we might call the economics and complexities scale. We face this problem from the beginning; that is, it is already a problem when we start the process of selecting equipment. It comes at us also in the management of the project, which includes the software applications programs, that we try to undertake.

In the selection of hardware one of the first problems is whether you are going to choose stand-alone equipment--that is, one for you and you alone--or whether you are going to share equipment with the computation center or some other group. Now, what are the advantages of sharing? It's very clear that you have a large, or comparatively large, operating group available to you. You also have a large array of processors and programming languages available. In addition you have a maintenance group and some kind of operating system. With the shared-equipment approach, then, you have a great deal more flexibility than you would get if you were developing all your own capabilities and interfaces from scratch. On the other hand, interfacing with all these systems comes back to haunt you because of the complexities that are introduced by increasing the number of processers in the system. So right here we see that there is a competition between the economics of scale and the complexities of scale. More on that later when we talk about operations and cost of hardware.

Let me say something about styles of interaction. The style of interaction between programmer and machine has a lot to do with the cost and rate of progress of your project, and with overall scheduling. Let us consider the following styles of interaction: batch processing, remote entry without text-editing, remote entry with text-editing, and interactive processing.

As you might expect, batch-processing will be the cheapest mode of interaction per unit of operation. It has been common experience that a project will move a little faster if you put on remote entry or remote entry with text-editing. The overall programming costs will not increase very much. You will find that as you go to remote entry with text-editing, for example, you will pay more for unit operation, do fewer operations; you will, in fact, be paying about the same for the overall development, but you will cut the development time somewhat. Before continuing, I would like to add that a number of installations are trying to get precise measurements of these programming costs.

I should further like to point out that management of the project is different for a remote entry text-editing development environment particularly with respect to documentation. In the batch-mode, things move slowly enough so that the programmers spend part of the time (between debugging runs) developing flowcharts and other communications aids. There appears to be a tendency for people working in a text-editing environment not to do this; I have observed, particularly when higher-level languages are used, that documentation suffers considerably. Applications programs are less well documented today than they were in the "good old days" when flow charts and program write-ups were supposedly prepared by programmers as a matter of course. I am not putting down higher level languages by any means. The point is this: it is easier for the programmer to understand what he's done when he uses a higher level language to do it; it follows, especially with the relatively rapid response in a text-editing environment, that more time is spent on getting the program written faster, and less is spent on meaningful documentation. The result is, of course, that at or near the end of the project you have to go back and make up for all the documentaiion that was not prepared on an "as you go" basis. I have put a heavy emphasis on documentation and communication because I am oriented to general systems where you are trying to get this kind of information to a large number of people. If you are working on a little self-contained system, you can probably get away with a lot less documentation, but in the general purpose area, documentation is a most important consideration.

Another problem encountered when deciding on the kind of interaction to use involves how much of the software you will be able to generate.

That is, how much of the software will you develop, and how much do you intend to get from the munufacturer? If you choose a batch orientation you can always get a batch processing system from the manufacturer. If you want a remote entry or interactive system, chances are that you will have to do a lot of the development yourself. Now, the prospect of developing your own system is most attractive; if your staff if big enough, you may be able to do this. But even here you can see problems developing down stream. For one thing, the machines themselves are being continually changed. Manufacturers continue their development of a machine after it is installed; these developments might be made for reasons of maintainability or to permit installation of new kinds of equipment. Changes can continue over a period of years. If you develop your own software you will either have to reject any given engineering change or change your software to accommodate it. Suppose you decide to reject the change. That might be an easy solution for the time being; what happens a year later, say when you wish to add a new piece of equipment that is dependent on an earlier engineering change? If you really want or need the equipment you must now make the retroactive hardware change and, in addition, modify your software. This can prove to be very difficult indeed.

Suppose that you go along with the manufacturer's software. Well it is still not uncomplicated because his operating system is going to change periodically. But he will try to develop an interface so that your applications programs will run from version to version of his system. Let me point out that if you go through a number of operating systems during the developing of a piece of equipment, or rather a system, you might make as many as a hundred different changes; this is going to require a lot of re-programming and all the rest, so the decision to build your own software or stick with the manufacturer's is complex and important. I'm sorry to say that in the end this decision is not often made on purely rational grounds. Very often you go along with what a colleague is doing or, more often, you go along with what the computation center, with whom you interact, is doing, You seldom have full control over this development yourself. But it is a question you should realize requires study and attention.

The next problem is that of scheduling and operation. My experience shows that an applications group interacts one way with an R & D computation center, and quite another with an administrative data processing center. If for example, you are interacting with an administrative data processing group, you will find that you are faced with a relatively rigid type of operation. They are not as elastic, say, as a research group or a student group. At the same time, because they are more rigid, you can fix a more definite schedule. On the other hand, they are usually in direct competition with you for that part of their operating day during which your programming group would like to use the machine. By contrast, suppose you are interacting with a research oriented computer center; here you would find that the users--the students and research people--are more flexible in regard to using the machine. You could probably re-arrange your computer run-times so that you do not directly compete for the same time-slot.

There are actually three groups that have to get on the machine: the hardware engineers, the software maintenance programmers, and the applications people. As we look at increasingly large systems, we find the hardware and software maintenance tasks taking large chunks out of the operating day. With a small machine, one in the hundred thousand dollar class, you can get by with a few hours a week maintenance; perhaps you can squeeze a lot of that into the week-end. But on very large systems, you will find that hardware maintenance alone can require three hours a day. Software maintenance takes another hour plus and your operating day is really cut into. Let me point out that machines are not yet designed in such a way that the hardware maintenance can be done concurrently with other regular operations. The engineers have to run special diagnostic programs to check out all the processors, for that reason they must take over the machine *in toto*.

As a case in point, the Stanford Computation Center's computer system requires an average of two and one half hours of preventive maintenance daily. Then there is about an hour and a half's worth of software maintenance in addition to that. Let me point out again that we still lack really adequate models for our software systems, and we rely a great deal on experience to effect the fine-tuning development of our systems. We do not as yet have sufficient conditions to guarantee that our software is "correct." This means that we are frequently turning up bugs, and in this debugging process, which can continue for years, we end up making modifications to our software system. This ties in with what I said earlier: self-change is at once a reward and a retribution.

So we see that very large systems mean large chunks of maintenance time. When we talk about trying to form large public utilities out of one or more large machines, we must realize that the scheduling of these mandatory maintenance functions is going to diminish the economics of scale that we hope to achieve by having the big, powerful machine in the first place.

At this time, let's take a look at some computer hardware prospects. One of the things heard repeatedly from the manufacturers is that computing is going to get cheaper as time goes on; frankly, I do not look for any real economies here within the next ten years. Consider the last generation of computers, characterized by the IBM 7090. This computer showed up around 1960 and was on its way out in 1967. There are still a lot of them around, but as a "generation" they lasted for seven years. When you look at the difficulty experienced by many users who are trying to get into third generation hardware, you can conclude that the current generation will be around ten years or so. Setting 1967 as year one, I figure that it will be 1977 before the next batch somes in. It is my observation that as IBM goes, so goes the industry. You can talk to Control Data or to Burroughs or to any of the rest, and they will be very candid about their position with respect to the "leader." This suggests that we will be stuck with this line of equipment for a few years, although this might be a blessing really. Many of you are probably aware of the trauma involved in getting into the third generation; a lot of managers do not relish going through all this again soon just to get into the fourth. It has been facetiously stated that project managers will not let the fourth generation in the door until they are promoted, leaving the heartaches to the follows who take over the line responsibility for getting the new generation computers on the air. Well, I don't know about that, but the statement is an indication of the magnitude of the pain. I believe that we will not see too dramatic a change from third to fourth generation machinery and software. IBM is working more in the direction of extending and improving their current line. Other manufacturers have few thoughts beyond their current generation.

What you are probably more interested in, however, are the prospects with regard to large files, mass storage, and terminals. Again we have been told that terminal costs are going to go down dramatically, and that we should all look forward anxiously to this. They will go down, but I do not think dramatically so. The companies that are getting into the computer and terminal business are still young and are trying avidly to develop their talents. Cost of the basic components are not going to diminish dramatically, and many of the small companies are still unsure of their markets. The teletypewriter part of the terminal business looks pretty stable; I don't see any reason for these prices to go down very much. For graphic terminals I think my point about the market being unclear holds, and I don't see any dramatic change downwards here. As a guess, the prices could go down perhaps 50 to 100% over the next few years .

A related consideration is that of communications costs. All that can be said here is that we are going to be faced with communications costs in the development of non-local systems. Currently, communications costs are linear with transmission distance; since this is so there has to be some optimum geographic distance over which you can operate and beyond which your communications costs will exceed operating and local equipment costs. This would suggest that a number of optimally placed regional centers would be more economical than one very large national center. Of course, communications costs can change, and the linearity argument could be removed if we were using some kind of special orbiting communications satellite. But for now, I feel that the costs, even if they become lower, will be essentially linear with distance, and we are still faced with determining the optimal size of a network of regional information processing centers. One thing is clear, I feel, and that is that we cannot arbitrarily communicate across the entire country without being overwhelmed by the communications cost.

Turning to the mass storage area, I am afraid that the picture is not too good. Experience shows that we can very quickly saturate virtually any storage device you can get. If you really want to have everything available to you, you think in terms of stacks of magnetic tapes and associated drives. This, of course, is the most expensive mass-store, with the cost per bit-to-be-accessed working out to something like five hundredths of a cent. This will be about halved when you go to something like an IBM 2314 disk drive. The cost per bit with a photo digital store will be about .00015 cents, with a much slower access time, of course. The photo-digital store is the cheapest (per bit) device manufactured, but it is being taken off the market for lack of interest. There are currently only two in operation: one at Lawrence Radiation Laboratory in Berkeley, the other at RadLab, Livermore. The photo-digital store works with film chips. The machinery is designed to do photographic processing, and stores the developed film chips in cannisters. These cannisters are individually accessible and are mechanically transported to reading stations where the chip can be removed, read optically and then replaced in the cannister. If you desire to rewrite information, you go through another photographic development operation.

Thus, the photo-digital store is essentially a slow-writing, slightly-faster-reading device. It has a capacity of 10^{12} bits, roughly equivalent to 20,000 magnetic tapes; so you see, it isn't really all that big. I'm sure that certain of you are already facing storage problems of that size and even greater. In fact already I have the problem at the Stanford Linear Accelerator Center. Out there our experimental physicists can load up 20,000 tapes in less than three years, so we have a very real interest in mass-storage devices. I might point out that, in addition to the mechanical-monster aspects of photo-digital storage, we must also consider its price, which is high, and the fact that there is limited experience in its use. These considerations are, however, academic since manufacture is being discontinued.

Let's spend another minute on tapes. You have available about 10^{10} bits of storage per unit; you can see that in order to match the photo-digital storage we just spoke about, you would need a hundred of these tape units--a football field full!

Incidentally, I have a limited, but useful, measure in this area; surely you have more precise calculations than the following, but to make a point to a group of students I once calculated that to punch up every character in a two million volume library would require the services of a Rose Bowl full of key punchers for one year. That works out to one hundred thousand man-years of keypunching. So we could say that the Rose Bowl full of keypunchers constitutes one unit (of information); some of my students promptly named the unit a Miller. Anyway, the point is that I believe that the development of information systems will tend to be more local discipline oriented systems than out-and-out, all-encompassing general, library fact retrieval systems.

That brings us to some remarks about cost and performance of systems. There are five components for developing an information retrieval system: conceptual design, systems design, user program, hardware related matters, and documentation and administration, by which I mean administration of personnel matters, space, general project costs and so forth.

While we are on the subject of personnel, I point out that in the conceptual design stage you are going to need high-level people; by this I maan people who have four or five years of demonstrated proficiency in software design work, particularly overall systems design. They are not easy to find. In fact I will predict that we will fall short--far short--of all the expectations and ambitions of people in this country regarding the development of information processing systems. It boils down to a poeple bottle-neck; as simple as that. We are simply not able to produce people of requisite quality and competence quickly enough to fill the stated needs and stated goals of many of our information sciences people. So keep in mind that you might have to modify certain developmental goals for the very real reason of scarcity of qualified systems designers.

In the systems design area the conceptual design is concerned with data management, storage design, and what you are going to do with the project; the systems design is primarily concerned with the processing of algorithms and with the interactions of your system with the operating system in which it is embedded. The designers must understand the operating systems very, very well in order to handle the interaction, or interfacing problems. Of course there are the user programmers; these persons--you can think of them as applications specialists-- have to understand what is being developed in enough detail to tailor the applications for its use, and to provide feed-back information to the system designers.

As for the hardware, it seems reasonable to try to develop some degree of machine independence, at least with regard to a line of equipment. I think one very often finds in these kinds of development the kinds we are discussing now--that the developers will switch from one machine to another. If you start with your own stand-alone equipment, you are likely to have a small machine; later on, you might have to move up a size or so. On the other hand, if you start working with the central computer, you might find out that in the course of development you have an opportunity to continue your work on a different machine, perhaps a stand-alone of your very own.

My model for the development of such projects around a large laboratory--such as we have at the University--goes something like this: I think one should develop a rather complete, general purpose software-hardware system...a centralized complex of computer power. As you begin to define special functions of stand-alone size, you pull these out of the central complex: you start to specialize. You now have few of the complexities of scale; true, you lose some of the economies of scale, but only initially. As you develop within your specialization, you start picking upoon efficiencies attendant thereto, and there comes a time when the operation should be transferred to separate equipment. Knowing just when to do this is the trick. Anyway, throughout these developments you will usually find that people change hardware at least once, going either from a general purpose to a small special purpose computer, or building from a small stand-alone system to a larger one. The big point is that it is most desirable to design in as much machine independence as you can.

Documentation is necessary at every level and in all phases of design and development. You can have the best ideas and what-not around, but if it isn't nicely arranged and intelligibly written down and communicated, you really have nothing at all. I suggest a technical writer from the beginning; he should report to the project manager.

The administration of the project should not and cannot be neglected. People, even many gifted and experienced ones, do not as a rule administer or coordinate themselves, as many of you have doubtless learned. You need a project leader and he needs some staff assistance. The leader, or manager, must interface the project with the computation facility, purchasing personnel, publications, perhaps even plant people. In the instance at hand, there are the formidable tasks of technical and higher-level administrative coordination, both of which require considerable effort and attention in a project like yours. Systems design will be coordinated by the administrative or executive assistant; technical coordination is the responsibility of the project manager. Of course there is also the very real need for timely coordination of persons working on the applications and of things.

Now a word or two about environment. This is something over which one does not always have a lot of control. For one thing, people in the university have a tendency to use what equipment is available to them. As you have probably observed, it is important for your designers to understand as accurately as possible what the application is all about. Among other things, this means that the designer has to get out and talk to the user. He has to interact with him directly. This seems to be a "resource" that is not so readily sought out and used. The graveyard of many a system has been a design that is not in the context of the user's environment.

Remember also that in systems design you must consider the differences between stand-alone or general purpose interface. In the former, people must know the hardware cold, in the latter they have to know the operating system cold. The user programs depend upon people who know both systems and applications, with the emphasis going to applications knowledge. I re-iterate my preference for the hardware environment: I like to see the development start and take shape under the general system, followed by a pulling out of the special functions as you develop a fuller use and need in that area.

As for elapsed time, I can cite an actual real-time development project, one that was about eighteen months long overall. Conceptual design ran about a third of the time: five to six months. Now you must recognize that there is always feedback in an effort like this; this shows up during implementation and the reason is the one I spoke of earlier: the lack of complete models that describe systems that change themselves. The self-change of a program means that you do not--you can not--see all the side effects of various perturbations in design here and there. You simply do not have adequate models to define the side-effects of self-change. So during the implementation you are always running into the need for little practical things that have to be incorporated. Okay, a third of the time in conceptual design, and then the rest of the period, twelve months, saw the completion of the systems design. The users programs are developed on the way; their development runs parallel to the rest of the design effort. Documentation is continuous; it should span the width and breadth of the project.

The costs. Most of you know what it costs to hire people. The toaal for the project I have been talking about ran to about $200,000 over the eighteen months. About two-thirds of that was salary; the other third covered hardware: machine use, storage use, terminal use, and so on. That's rough, but it should give you a picture of the major break-down.

Well, this brings me to the end of the general discussion; I would like now to hear from you. As I said earlier today, I have been hit over the head with a lot of systems and I hope you will now take the opportunity to hit me over the head with some others, giving me the option, of course, of hitting back.

Discussion

Kilgour: Would you please comment about the problem of reliability?

Miller: Reliability; here again, the complexities of scale and the economies of scale are in competition. The reliability for you in operations is getting better, but in terms of the reliability present in real time, they've actually been getting worse. Because of the complexities of the interaction of the many different units, an increasing percentage of real time of the day is devoted to preventive maintenance; this, I think, is an example of the decrease of reliability in terms of units of real time.

At the circuit level, the biggest problem is still the interaction between the software and the hardware, whether the hardware and software know what each other are doing. In terms of just plain hardware reliability, I don't see a major change, but it's better than it was a few years ago and it's probably reaching some asymptote that will not change dramatically until we get into a new kind of circuitry for the main processors or until we get off the electro-mechanical devices for auxiliary storage. Now any of the kinds of auxiliary storages that one can foresee for the immediate future have some mechanical control in them, and that limits reliability. I don't see that changing very rapidly.

Kilgour: Given the requirement of an online library system not being down for longer than two consecutive coffee breaks, what's your estimate as to how you could fit with that requirement?

Miller: Let me tell you of a solution that was taken by U.S. Steel in the development of their rather large all purpose system. They're getting some Burroughs 8500's in. They decided to partition into two completely independent systems, each of them with two processors. They needed the reliability of a four processor system. But their partitioning was into two completely independent systems and then two that were linked. The critical factor here is that you need to partition the operating system. You need that independence of the operating system, and if you really want that strict a reliability consideration, I would certainly duplex or multiplex the machine extensively. I'm shooting off the top of my head, because I don't have that problem, but certainly duplexing the processors and some of the other equipment against the cost, I think this U.S. Steel solution is probably about the right one. I thought in terms of what I would do if I were going to make a large utility for Los Angeles, San Francisco, New York, or some large center, I would certainly go in that direction. I would have quite independent systems; they may share some equipment. If the disc file is completely down you share one of the others, but I would go for heavy duplexing and have relative independence.

Kilgour: You don't see, I take it, then, one hundred percent duplexing, but maybe one point something percent so that if part of it went down you could at least "limp."

Miller: Yes, "fail soft" sort of thing is what people talk about. In a normal system, for example, you have more than one disc anyway, so you've sort of a duplex there already.

Kilgour: Then the CPU is the real problem.

Miller: No, it's the CPU, the channels, and the operating system.

Kilgour: Your statement about the cost of communication being linear, in New England, they were dropped to a quarter with distance, including all of New England . In a small area, such as New England, and not long distances, then you're longer distances are down to a quarter of what your intra-state costs would be.

Miller: Yes. That would suggest, possibly dictate, a certain regional size. Satellites may change this. There are people, Ford Motor Company for example, doing time sharing all over the world, or practically all over the world; they do it in Germany and England via satellite. Incidently, they do time sharing for real money and I'm always impressed with people who do it for money instead of for fun. They do it all the way to England and Germany via satellite, and they don't have a linear cost. That can change the picture for us, but that's not here.

King: In talking about specialized functions that might be identified and implemented on separate special purpose machines, are you thinking about things like display control and editing capabilities?

Miller: No, I was thinking of a larger scale application. For example, in our Linear Accelerator Center we have a 360/91 which is just coming into operation. We went from the 50 to the 75 to the 91; that's the reason I have the feeling that people change equipment somewhere along the line. We have certain internal functions in the 91 which are not a good use of the machine. But they were a good way to develop. One of these functions is the graphics and film processing operation. We had a lot of film data to handle . Physicists can turn out four or five million photographs a year, and if you give them half a chance they'll triple that. We have film digitizers which operate off of the main machine. The digitizer itself runs like a tape unit really. You put the film in, it digitizes it over a frame, buffers that, sends it into the machine, and so forth. Now, on the 75 this was not badly matched to the operation of the machine, but with the 91, it's out of balance. That's a kind of function that I would pull out of the machine in this first sense that we're talking about, and have a little buffer controller outside of the machine which will do the controlling of this film scanning. A lot of the interim processing associated with graphic interactions should be done outside of the machine.

I think there's already a higher level kind of partitioning that I would do. As you get a sufficiently large use, the whole information system could be pulled out into a stand alone machine. Again it depends on the demand and what you expect to be doing with it, if you've got enough use. On a campus, some of the major teaching functions can be pulled out. If you're dealing with a sufficiently large demand for, say, one of the compilers that is going to cover a large number of courses, then you can pull that one out and put it on a special purpose machine. You find that it can be very economical to operate such a system. You've got one system and one machine; the maintenance problem is much less because it's not interacting with other processors. I would tend to work for pulling the whole application out.

King: Just a comment on this. I haven't looked at this, but for an information system one might conceive of operating it, as I think some people have, in the machine with a large memory using a partition which was allocated to them on a semi-permanent basis, and under an operating system in which they use the CPU relatively modestly because it's a very I/O bound problem, they may have a dedicated channel and some dedicated storage device. Looking at this kind of environment, with all of the systems working and all of the hardware involved, and the possibility of duplicating this on a special purpose machine, you would need a large file, an interactive operating system of some kind, and communication controllers. It's not the sort of special purpose machine that costs 75 to 150 thousand dollars that people are used to at universities, and it's a very large scale system. It's an order of magnitude different, I think, than the kind of special purpose facilities that are ordinarily suggested.

Miller: In theory, you have to have a large scale before you begin to do this, before it begins to make sense, but when you're talking about a very large library, with the bibliographic and administrative operations necessary, you're coming into that scale. You're going to find yourself renting a sufficiently large number of central machines. Things like files are completely linear. There's no economy of scale; you're going to file on the big machine, and run on the small machine. You have the same costs, and when you get to the place where your disc files, or whatever your auxiliary storage, are an appreciable fraction of the cost of your system, it doesn't make any difference whether it's on a central machine or not.

King: I just have the suspicion that it might very well be cheaper to add a channel and a couple of files on the big machine.

Miller: I wouldn't want to mislead people; it shouldn't be decided at that early stage. There's another advantage, particularly in the early development of a system, of doing things on a central complex. One should never underestimate the advantage of this sort of central pool of intellectual knowledge. I mean, you spend an awful lot of your time going and asking people how things work, how things function. The pooling of the various simple little processors that people make, channel control programs, editing programs, the input output programs, and so on. One should never underestimate that. There comes a time when, if the system gets big enough, the kind of operation demanded by the information system is different from that demanded by the central system. You're in a much more rigid situation, and rigidity of the information system is much greater than, say, that of student operations or research operations. The research man is always so busy that he's got something else to do. He's pretty elastic. He's off the air one day, he does something else when he gets back on. The library system is quite different from that, and now you may need to freeze your system. These considerations drive you towards this more special solution.

Kilgour: King's suggestion would work locally completely stand-alone, but in a network, as a node in a network it would be difficult to invoke that solution, because, having a 91 in every node in which you can sit and find a core is unlikely. It depends on what kind of system you're looking at, whether it's completely stand alone and local, or whether it's going to be a node in a network.

Miller There are a lot of things that we don't know, unfortunately. We don't really know how to characterize the dynamics of a program, and therefore, we don't know how to charge for them, and since we don't know how to charge for them, the economies of some of these things are a little elusive yet. People suggest that core residence time is perhaps the most important thing to charge for and in some systems it is. In our system, we're dealing with a 91; it's not so obvious that that's the right way to charge, but it's clearly an important part of it. That will dictate whether or not we have people sitting there, loss of revenue and that sort of thing.

Stevens: I appreciate your opening remark about Allen Veaner asking you to tell it as it is. I've heard him say to me some of these same reservations that have come up as Stanford has found its way into using computers for Library technology. As we at M.I.T. look toward the same problems we're also faced with some of the realities what you are coming up against. But I file a reservation with you and ask you for a rejoinder on the grounds that some of our experiences are different from yours. Particularly your figures for maintenance time; I've experienced times when the machine has been down for two days and if I divided that over a week I run into the kind of maintenance figures you're talking about. But as I think about the realities of maintenance and machine technology, where machine technology means computer machines, I have to look back in history to see what happened, for example, when the diesel locomotive came on line, It's early history of maintenance was that five hours of operation led to twenty-five hours of maintenance; the diesel locomotive now runs for something in excess of 6,000 hours before they look at it and see if it needs oiling. The same thing happened in jet engines. The very idea, as you put it of "fail soft," led to the development in air transportation of the twin engine aircraft. Two engines were indeed needed for take off, but once you got going you could fly with one. The same thing, I think, may indeed happen in computer technology at least in university applications, where you take off with two, but the program can then fly on a "fail soft" basis with one.

Miller: You buy yourself reliability in that fashion by duplexing and multiplexing. But I'm not sure that I understand your conclusion well enough to try to construct a rejoinder.

Stevens: I'm not exactly asking you for a rejoinder. I'm asking you in this one instance whether you can see, as I do, that downstream from where you see it as it is, where you see it as it will be, that reliability will go up markedly. History tells us that it will because the customer will demand more reliability.

Miller: If customers demand it, we can get it through multiplexing, in that fashion, Customer demands may not make it possible for us to get higher reliability on the same device.

Stevens: No, I don't suggest it's going to be the same device. I simply say we're going to achieve the same end.

Kilgour: This is of course what happened in the diesel locomotive because you have a system and you shut down on one unit and the supervisor gives it the once over while you're on the road. Your maintenance time doesn't exist in the terminals, and this is the kind of thing you're talking about. When you're down hill you can take a unit out and do some preventive maintenance on it.

Miller: I don't know how to argue from the analogy actually. Things develop more slowly in time. We have increased the reliability scale in the last ten years, but I don't see us getting away from the mechanical devices for auxiliary storage, and those are really close to their limits.

Stevens: May I pick up that point? The work that I've seen going on a Lincoln Laboratory at M.I.T. and also M.I.T. proper point out to me that what we've seen in photo-digital memory so far may only have opened the door to that whole concept. We're beginning to hear about work in new storage devices that will allow us to do the kind of storage that costs less than you have indicated and will be commercially acceptable, where others have not been acceptable.

Miller: I know, but I don't expect them for ten years.

Stevens: I do.

Miller: You do. Good! It's true that one tends to overestimate what you can do in five and underestimate what will go on in the second five, fortunately for us in terms of progress. You know, it's a long way through mauufacturing marketing, and cranking up your system to accommodate them. We may see it in the second five years. I'll be glad if that happens. I don't think we'll see it in less than that.

Stevens: My view is that the path gets shorter and shorter, and, without trying to monopolize the question session, I'd like to add another point. With regard to your concept of machine organization, wherein you state that a dedicated machine working out administrative problems can get its job done because it is dedicated, and one where students are involved may take a sort of second order system of ordering their work. At M.I.T. where we have a good deal of computing facilities for students we have the greatest computer famine in the nation, and it's not because of the administrative programs that are running on those machines, it's because the students come equipped to use those machines at a rate and with an intensity and with a fire and drive to persuade the administration that they've got to have first use of those machines. If the pay checks don't get printed, well--

Miller: Again, I'm not sure what your point is.

Stevens: My point is that students' use of machines is going to push us so hard that we're not going to be any longer in the arena that you tell "it as it is," where administrative uses of the machine take first priority in an academic age.

Miller: The demand for that access time is going to force a different pattern.

Stevens: That's correct.

Miller: Well, OK, we're certainly seeing some of that here as elsewhere, I'm sure. On the other hand the society, unfortunately, is still oriented toward a working day mode, and it takes a long time to change society to work on a different model. People have proposed-this is a digression, but it's not irrelevant- very dramatic things in a 24 hour model, but this is not easy to do. It seems we have three opera companies: all of the things that have been partitioned into night and day have to be triplexed if we're going to work on a 24 hour model. I don't see that happening quite as easily as you think. I think there's some inertia there.

Reimers: I'd like to disagree.

Miller: OK, that's good.

Reimers: In the Washington area we've been discussing announcement of the fourth generation computers as of about 1971. Let the records show groans from the audience.

Miller: Well, maybe. What do you think of the fourth generation?

Reimers: Well, this is going to be a fourth generation: it's going to be mainly in the main frame; it's going to result largely from the large reduction in cost of integrated circuitry; it will probably have the majority of logic in operating machines, and this is going to give a large answer to your reliability problem.

Miller: This generates of course, again through redundancy in technique, a very large increase in reliability. I'm always ambitions and eager to see new machines come along, but my feeling is that the acceptance rate of those will be relatively slow. It's not that they're not available. My prediction is based on inertia generated by the investment in current machines, the turnover problem. That was my prediction.

Reimers: I expect machine organization is going to remain about the same.

Miller: It wan't that they couldn't be made available. I mean we could turn them out; I could point to several places where a fourth generation machine is available tomorrow. It's the inertias of the organization, I think, that will prevent their acceptance. I don't think that puts us at a disagreement here.

Campbell: What do you think of or what's the general reaction to, the Wall street Journal reports regarding things like anti-trust with IBM? The tie in of sales of software and hardware. What effect will it have on all this?

Reimers: GSA, General Services Administration, is also moving in its government contract awards, trying to divide software and hardware. You have this coming from two directions.

Campbell: Well, will this affect fourth generation equipment, and what is your reaction to this?

Well, as a sort of system designer, I would hate to see hardware and software separated. I think they need to be developed together; in fact, generally I think one wants to develop the software first. Actually, the optimal thing to do is to develop a system and decide what you can do in hardware and in software as you go. I guess I really have no feeling as to what it will do to the market if they are separated. I don't think it will encourage the turnover of equipment. I think it's somewhat important that one develop a system and then decide on the partitioning of that between the various options that you have--hardware, software, the sort of things in between, like microprograms. Microprograms sort of come half way in between, and particularly the operating system. The operating system is the machine that you see; do you think of that as really separate software? That's the machine that you see.

Campbell: Could I ask Mr. Reimers to explain why the GSA is interested in this separation?

Reimers: They feel that they can get greater economies overall because the government, is moving toward purchase rather than rental of equipment. They feel that they can purchase software and they can develop the operating systems more cheaply, and come out with less money to the taxpayer overall. They also hope by this to get a modularity in operating systems which the manufacturers will not give.

Miller: I know these ambitions. I don't know what to comment. I'm on the skeptical side.

Dix: May I come back to the chart on the screen? This is of great interest to those of us who are here -- you present this primarily, as an area of managerial decision rather than details. That is exactly what we've been asking for, a lot of us, but I'm somewhat concerned about it's apparent precision. In the little right hand corner I've got a nice neat budget. What is this supposed to buy for me, in terms of comparison? [Laughter].

Miller: This was a development aspect of the information system that was a modest sized information system, not that one that was intended to incorporate all the functions of the library system. Dick, exactly what function of the system does this represent?

Bielsker: This primarily involved a very sophisticated text editing system so it was not too involved in the area of retrieval itself.

Miller: But the figure was intended to include personnel costs and machine costs of development, but not subsequent operating cost. Well, of course precision--you know, you spend more in one area and less in another. No management will say that you can plan and budget precisely, but you lose none of the planning value with rough breakdowns which I think are ideas for guide lines. I've developed a lot of systems. I find that this approximates the developing and partitioning of functions between conceptual designs, systems design, user application, and so forth. Smaller systems will cost a hundred thousand. A simple compiler system, well specified from the beginning will cost you easily a hundred thousand dollars to get on the road and a very big system that interacts with a lot of other people will go up to a half a million. If you go up to a general purpose operating system, the kind the GSA is fighting, you're talking about a thousand man years of effort. This is forty million dollars of cost. There are other figures that are useful. If you want what it costs for a programmer operating in sort of a natural environment of having machine time and so on, some clerical support, secretarial support, you'll spend between thirty and forty thousand dollars per man year of effort. That's cheaper than hardware. I think an experimental physicist will run you seventy or eighty thousand dollars per man year. I don't want to pose everything too precisely, but it gives you sort of the general idea of the things we have to deal with.

Hammer: I'm faced with the everyday problem of a working system; when you get to the people running it: they have the best sophistication. When are we going to give the people a chance to catch up to the machine? And the software a chance to catch up?

Miller: I stated earlier that the training and developing problem was the country's major problem. I think there are not enough trained people to meet the ambitions of the country. I think that will continue to be the case. The problem comes back to the simple, intellectual problem that we don't know how to engineer systems that change themselves. It's pretty hard to teach people; where you know a lot about the system, you can teach them a few of the principles that go with the work. Here we have to get it by hard knocks, by experience, and that's a kind of linear way of building up experience, rather than a more exponential or quadratic way if you've got some better principles, the people are coming to models of operating systems and models of dynamics of programs, but this is several years down the pike. I mean this is not going to help you tomorrow.

Hammer: Unless you can get the ideal--the software to take over.

Miller: Thank you. You will spend four or five years building up a good laboratory of people today. I seldom build a group in less than five years. In five years you build a group that can take a machine and buff it and fly it. But that's a long, slow period of development.

Shoffner: If you build a seven man group of that sort and you pass through this project and pick up another project, and have some turnover of personnel, with such a group size are you going to be able to run a reasonably good shop over a ten year period? In other words, is it a continuing developmental group, or are you depending when you talk about this project on drawing your personnel from a larger pool of people?

Miller: Certainly these people are already well trained, and, in fact, you know how they're going to fit together. That is the assumption of all continuing working groups. One of the problems in building up your laboratory group is that the people who are well trained in certain areas, still have to be put together, and this assumes that they already fit, that they know how to work together.

Shoffner: The reason for raising the question is many of the libraries are putting together groups that are three and four man groups, and I question whether or not this group size is in fact large enough that they can maintain a group continuity with personnel turnover.

Miller: It's bordering on the possibility. I would suggest that such a small group ought certainly to be in the context of an experienced, larger group that can buffer it over this problem and an experienced larger group around it standing out by itself will be fairly formidable.

Hammer: There's one item that I don't remember your mentioning and that is manufacturer's honesty. You may not want to comment on this, I don't know. [Laughter]. But I know of nothing more exasperating than a claim for some piece of equipment that doesn't live up to actual performance.

Miller: This is another aspect that's taking four or five years to build up your laboratory. You have to have good people who know how to examine a piece of equipment and not take the salesman's word, so to speak. I have not had much trouble with manufacturer honesty in the last ten years. You usually throw a team of people into the investigation of the equipment; it calls for our own evaluation of our own information about it.

Spaulding: I think that it's a matter of manufacturer's reliability perhaps more than honesty. There is a problem even if you don't believe anything they say. Dr. Fussler had software and hardware problems that were not as the manufacturer represented, and simply taking the position that you didn't believe any of it wouldn't help. What you have to do is completely check out the entire software system to know that it wasn't going to produce that way, or, in the case of his terminals, he would have to run them for quite a period of time to find out that they were not going to produce. And people concerned with, say, a data cell, would not know if it was going to take a beating until they put it into service.

Miller: The data cell is a good example. If you study the initial announcements, plans, and descriptions of the data cell, you're not misled. It was not intended for what most people thought they could get out of it. I think in this case the users have to share a fair amount of responsibility for letting themselves be misled. If you look at its structure, it's more like a tape unit than a disc, so that if you use it in that fashion, it's not an unreasonable device. We went through that very problem. We very nearly misled ourselves as to what we should expect from the data cell. We had a couple of data cells on order; partly through other people's bad experience and partly through our re-examination of what one should reasonably expect from it, we decided to cancel out on them and didn't get them. I thought that we perhaps misled ourselves a bit, and I didn't really feel we'd been misled.

Spaulding: But it did take a sizeable, skilled group to determine this, whereas under normal circumstances there would be the librarian and the salesman, neither one of whom would probably make a very good evaluation of those circumstances.

Miller: I guess it comes back to the point that you shouldn't underestimate the value of a pool of intellectual knowledge around a central operating system or some soft of computing system. These are people that can help you evaluate devices.

Rogers: Dick Bielsker appeared to be a silent partner, but actually he did participate in the paper I understand, I believe some people have misunderstood the chart, thinking that this represents a library type automation project. Would either Dick or Bill care to comment on that?

Bielsker: I think a regular library system would be triple of what we saw on that chart. What we're talking about there is the kind of library system that we're going to do as a follow-on to SPIRES and I say "follow-on" because this chart really reflects some experience and background building a prototype of a similar type system. It's not based on just charging into an unknown.

Stanford's Data Link Network and Display Terminals - What They Mean to the University's Information Retrieval Projects

Roderic M. Fredrickson
Associate Director
Stanford Computation Center, Campus Facility

Mark D. Lieberman
Assistant Director
Stanford Computation Center, Campus Facility

A Paper Prepared for the Stanford Conference on Collaborative Library Systems Development, October 4-5, 1968

Partly in response to the stimulus of Professor Miller's remarks, I'd like to start by briefly outlining the magnitude of computing services provided to the Stanford University community. This background will aid in evaluating the setting in which a library project like SPIRES/BALLOTS gets its support, and how research projects and the Computation Center interact.

First, we'll take a look at the Computation Center itself. It's composed of three facilities. First is the SLAC (Stanford Linear Accelerator Center) facility, that Professor Miller already alluded to, located several miles away off Sand Hill Road. This facility is managed by the Computation Center under contract to the Atomic Energy Commission. We have recently completed installation of a 360/91; it gets turned over to the customer, us, on Monday. It's already been operating for the last couple of weeks. The primary purpose of the SLAC facility is to serve the needs of high energy physicists in connection with the experiments they conduct on the linear accelerator. Typically, their jobs run longer than most other computing tasks at Stanford. During the period when the accelerator was under construction, physicists were served off the central machine on campus. You see here an example of specialization in equipment, where longer jobs are pulled off from the central university machine onto a separate machine. I don't mean to imply that's bad; I think that's good, and that's probably the direction things will continue to go.

The second major installation of the Computation Center is the Campus Facility. This is a 360/67. This machine provides general purpose computing power for the university community, for research, instruction, teaching, and so on. It provides batch service, text editing and remote job entry, and as of Monday, October 7, provides time sharing. Now within this particular facility, a prototype library system is being developed; in a minute we'll expand on that, and show how it's fitting into the system, what's good about that, what's bad about it, and what we expect because of the way it is fitting in.

The third facility is called the Real Time Facility. It is another example of specialization in computing at Stanford. This facility was developed to try to meet the peculiar needs of the hospital and the Medical School. Emphasis here has been on development of facilities for real time data acquisition within the Medical School. For this purpose, a time sharing system on a 360/50 has been developed. This machine has two million bytes of bulk core, with an 1800 tied onto a channel adapter to provide for data acquisition.

These three facilities are under the directorship of the Stanford Computation Center. For each facility there is an Associate Director, and a staff, and we all report to a central Director. The budget for these three facilities is on the order of 5.4 million dollars a year for operations. The Computation Center includes equipment and operating budgets. Hold this picture in your mind for a moment, because it becomes more relevant as we go on; I think it's an important step that Stanford has been able to pull facilities together into one administrative body. I'll argue a little later that they haven't gone far enough, in my opinion. One of the reasons we think they haven't gone far enough comes out of our experience trying to work with the University Libraries in meeting their computing requirements.

Let's describe how SPIRES and BALLOTS fit into the current Campus Facility's operating system. I'm speaking now from inside the systems, as opposed to Ed Parker and Al Veaner who speak at various times outside the system. To me, they look as if the're users. I'm sure they feel they want to cross over the border every once in a while.

Our mission is to provide general purpose computing power to the university community. We have tuned a system that's designed to meet a particular kind of work load that exists within our university community. We need to accommodate a quarter's mass influx of students, with thousands of jobs a day. We've got researchers who've got to get their work done, too; sometimes they have jobs that run as long as a half hour. We no longer have on the central machine the really, long grinding jobs, since they're now on the

360/91. (Typically, physicists have been the ones who've gobbled up the machine time, much to the consternation of the little guy).

Chart 1 shows the hardware configuration of the Campus Facility machine.

The system utilizes the manufacturer's main operating system, OS. For those familiar with OS jargon, this is an MFT system, version 13. Language processors have been brought up from other releases so that users deal in terms of the latest version of the language processors. We have modified OS some; I won't go into the details of those modifications.

The first level of software within the system has to do with support of batch services. We have taken the HASP system, imported from Houston, and of the original 12,000 lines of code have changed or added about 6,100 lines in tailoring it to meet the needs of the Stanford University community. Under the STANFORD/HASP monitor, we schedule two types of batch service. The first, "Production Batch", is oriented toward the researcher. The second, "High Speed Batch", is oriented toward the student user, though some researchers use its facilities as well. In addition, the STANFORD/HASP monitor also controls through the plotter partition, on-line plotting facilities.

The line connecting HASP and WYLBUR represents facilities by which terminals may enter jobs into one of the batch partitions, retrieve "printed" output for review at the terminal, and inquire about batch job status.

The second level of software is concerned with support of terminal services. Starting with MILTEN, terminal I/O is routed back and forth between WYLBUR, the text editor, and ORVYL, the time sharing monitor. ORVYL is designed to support not only installation-provided processors, such as BASIC, but also user-written, time-shared interactive programs.

A special feature added to MILTEN was RCP which permits user written code residing in one of the batch partitions to communicate with terminals. This feature was added in support of the SPIRES/BALLOTS prototype development effort.

Now SPIRES/BALLOTS is a core resident body of code which receives its terminal services by communicating through the MILTEN supervisor. The Campus Facility's role in this has been to provide those systems services which will allow the SPIRES/BALLOTS staff to prepare "applications code" that will allow them to communicate with terminals at the same time the rest of the system, i.e., WYLBUR and ORVYL, are communicating using terminals. Other than maintain the integrity of the system, and provide these services, our job has been relatively minimal. SPIRES/BALLOTS is writing their code in PL/1, which in itself is no mean task, probably a project in itself.

Early estimates of storage requirements and the critical problems we already have on the system -- the amount of I/O that goes on -- made it necessary to install additional channel and disc facilities to meet the library project's requirement for prototype operations. The Campus Facility already had three IBM 2314s. These 2314s include all user files connected with time sharing, text editing and batch operations on the system, i.e., some system residency is included. There are two drums for frequently used system components, for paging functions for the text editor, and for the time sharing system. In the original configuration there was one selector channel for two 2314s and another selector channel for the third 2314. So a third channel was added with a dedicated 2314 for the library project, which represents a sizable investment. Yet our cost analysis so far still reveals, at least for the prototype system, this to be the most economical way of providing computing capabilities to the project. Probably in the long run, that won't be true.

Services available to the library project are those available to any standard OS job with the addition of communication with terminals by calls on the services of MILTEN. A number of new services have been requested by SPIRES/BALLOTS projects, services which either are not needed by other users of the system or which are ahead of our intention to provide to the general community of users.

One such service is in the area of data management. Data management functions at the level of SPIRES/BALLOTS is strictly a function of what files they can allocate as part of normal OS operations. We're being asked to give some consideration to providing for a more dynamic file capability, that is opening, closing, and manipulating files, other than simply when the OS scheduler is available. This is a service we have provided our own supervisors, so technically we know how to do it. The code, however, is quite sensitive to misuse and can have devastating effects on system integrity. We are reluctant to extend this service to code not under our control. This example illustrates one of the kinds of problems that has to be faced up to in this type of application, in this type of environment.

Another pressure point is in the area of CRT type displays. Here SPIRES/BALLOTS would like us to move much faster than we feel is healthy for the general development of this capability.

Let me give you a little bit of history about this, because I think it's an illuminating one. Originally, the thinking was that the IBM 2260 would provide the text graphic support needed by the library project. We support currently only IBM 2714s. We installed some 2260s on the system on a two-fold experimental basis. First, it was experimental to us, and second, it was to meet a commitment to the library project for text CRT support. The 2260s seemed the most reasonable way of doing it. We didn't use the manufacturer's software, but wrote our own, partly because we already had a supervisor structure for communicating to remote devices, and manufacturer's support wasn't appropriate for us at that point in time. We brought it up, tied it into the then available text editing facilities, and made some changes in the way drivers worked, to try to correct for certain human interaction problems that we found, given that the original software was designed for 2714s rather than for 2260s. We used it for several months before letting the Library look at it. We weren't too happy with it, and they were even less happy with it.

Several different things were really wrong with it. First of all, we couldn't maintain any decent response time on the 2260's. For what the Library wanted to do, it took about 3.4 seconds per tube to do a full scope regeneration. We only had eight 2260's on the machine at that time. The Library was talking about many more than that, so there was really a major response time problem that was a function primarily of hardware architecture of the 2260 and the 2848 control unit. Other problems had to do with the fact that the characters available on the 2260 were not extensive enough to support adequately the requirements of the Library, and this, of course, we knew when we first started the project. We had decided to live with the problem, but it became more critical and more unbearable, as the library project itself developed its application further. They became more anxious for expanded character capabilities. Also, I think they became more sensitive to the problems of the human factors involved in using the 2260. This is my opinion; they could comment best on their own.

This experience illustrates a typical hazard for administrators not technically competent in computers. In the beginning you're so concerned with getting a project going, that you tend to focus too much on the mere technical problems in getting it going, saying in the back of your mind, "Well, I'll solve the human factors problem when I see it on the terminal." And it works pretty well, except when you've got some severe limitations built into the hardware -- then you're really trapped.

So after the 2260 experience, we sought different approaches to providing text CRT support for the Library. We looked at a number of different devices and facilities. We came up with what we've termed "middle level" CRT graphics for this information retrieval project. We see on the market today devices at two extremes. One end of the spectrum is best typified by the IBM 2260 or the Sanders 720 - a straight character generator-oriented CRT facility, limited in its character set, limited in display format to a fixed number of lines of predetermined width. At the other end of the extreme, we see the capability typified by the IBM 2250 or the ADAGE machine, which has extensive graphic capability. But now we're talking about considerably more money than we can justify for a number of terminals in public use. In addition, with these very powerful graphic terminals, there's the substantial technical problem of communicating effectively over long distances. They require bulk data transfer rates and coaxial cable. But middle level graphics might provide more flexibility than at the low end of the spectrum, and something that would fall short, necessarily, of the full graphics approach.

One approach in this direction that we have been looking at uses a small disk for the storage of video signals. An example is a product offered by Data Disc which uses standard TV monitors at the terminal site.

Speaking as the manager of a facility trying to provide a service not only to the library project, but to a general university community, we have a responsibility for the integrity of the total facility. Most of the problems we have with the library project, or for that matter, with almost any user - the library is really no different - is the system's integrity. There's a little fence we normally build between the system and its users. It's very important that we maintain this fence. We have overall system functions and services, and here we have an interfacing capability for calling upon these system services. When you work with an information retrieval project, this fence gets chinks in it. Part of the problem is that such users need access to privileged information. It's not so much that the facility isn't there; it's just that on the system side, we may not have built in the necessary protection against a foreign user employing that particular facility. This does produce a reliability problem from the systems point of view, and it's probably the only area of difficulty we have in dealing with the library project. It's an area of difficulty that men of good will can work out. It's just that our responsibility is different from the library's project. Their responsibility is to get their project operational. Our responsibility is to maintain the integrity of the system for all the users, and sometimes those things don't quite meet. It means they sometimes have to go slower than they'd like to go, and we sometimes have to ask them to go a little slower than we'd like to have to ask them to go.

Let me now describe our terminal communication facilities and try to illustrate yet another problem we face in implementing the library project.

We communicate to terminal devices using IBM's 2702 Transmission Control Units. I think most of you are familiar with those. We've just recently installed a PDP-9, and in January we'll be installing the rest of a system to replace the 2702s.

We'll be bringing in all of the terminal devices through a separate stored logic machine, as opposed to the 2702 which is a fixed program machine. Now our goal here is to pick up some reliability that we don't currently enjoy with the 2702s, and also to provide a more convenient and reliable way for users in the Stanford community to tie on foreign devices of their own. Now the software support for the PDP-9 will be interfaced to talk not to the OS portion of the system, but to talk to the non-OS portion of the system, those parts of the system that go by the names ORVYL, MILTEN, and WYLBUR. That's going to produce a problem for the library project, in that right now they happen to be running in the OS portion of the system. Whether services other than the terminal facilities of the PDP-9 can be provided directly for foreign devices (like CRT's having higher data rates) is still subject to study. The PDP-9 will be talking with ORVYL, and there is some probability that it will not be able to talk to the SPIRES/BALLOTS partition. But there are still ways in which the services of CRTs can be provided to SPIRES/BALLOTS but not through the MILTEN monitor. There's nothing preventing the library project from talking to the CRT directly as an OS service. We can provide the same kind of software that we provided for the 2260. I think Ed Parker, or perhaps Al Veaner could best speak what motivates them to want to go in the more general direction, but I'll leave that to them.

What is strange in this situation is that the library projects would like us to establish a campus standard for CRT devices and support which they could use as a part of their project. This shows a sensitivity to community goals that is rare among users. The introduction of general support is a much more difficult thing to do than simply supporting a device for a particular project. A consistency of service has to be maintained for the investment in effort to be amortized through new use. This means that the device must be correctly fitted to the system both from the hardware and software point of view. It also means that the market for the service must be large enough to support the service at a reasonable rate. A few users, with special needs and funding, can make a new service appear economically reasonable to support. But what happens when their need changes or the funding folds? What happens to the smaller user who has been led by the service's availability to integrate it to his research? Can he pay the rate made necessary by the loss of the larger user? Not likely. Here I think our responsibility is clear. We can only offer as general services those things which we can reasonably predict to be marketable over a broad market at a stable rate.

As a part of the problem of meeting the library's needs, we very quickly discovered that it might be difficult for us to link other people into the system. We have out about 130 terminals on the system now; most of them are within the university, and some are a fair distance away. They come in on standard telephone facilities. Terminals on campus come in on what's called a data concentrator, using leased lines for the most part. We do that partly because we get a little better price, and partly because we picked up much more reliability by avoiding the switched network. Also, we're able to troubleshoot defective lines much faster this way. Now a major disadvantage of the ordinary phone line is that it doesn't provide transmission speed higher than that suitable for 2741 or teletype terminals. And this is a problem universal within the telephone company and computer users. It's not unique with us, but our feeling, contrary to the public belief, is that IBM is not the greatest impediment to remote use of computing -- the telephone company is.

Now, partly because of the trouble we have with the telephone company, and partly because we feel the common carriers haven't been sufficiently responsive to the variety of needs on campus, we've been trying to anticipate a requirement which I think will be mandatory in the future, i.e., there will be more and more direct (hard wired) links laid between the university's computing facilities (to the degree that they're centralized) and outlying stations. In the future you won't find a university built without large conduit facilities or without very careful siting of its central computing facility.

Someone was telling me the other day that libraries used to be built with networks of vacuum cleaner conduits in the walls. I guess that isn't done these days, but it's a good thing Stanford's

Main Library has them because that means there's a lot of wire pulling space in the building. I don't think a university ahould be built nowadays without greater attention to the problem of communicating over wires between outlying stations and the central computing facility. The problem we see in developing this kind of network is the difficulty of paying for it. Over the last year and a half we've spent roughly fifty thousand dollars of the user's money in trying to establish the rudiments of a network. We've connected the Medical Center's Model 360/50 with the 360/67. We haven't yet found an economical way of tying the 360/91 into the network, but hope to solve that problem eventually. Right now we're pulling lines between the 360/67 and the Library. We already have lines to the chemistry area and the Medical School. We have some at the Electronics Lab, too. All this is just a start, and it's much too shortsighted.

Our present estimate is that it's going to cost about half a million dollars to adequately provide for inner communication between various parts of the campus and its central computing facility. We're not really doing too much about it now other than trying to articulate the problem. Some of the people in the community don't yet recognize this communication problem. For example, we see a lot of small computers - machines under $8000 - that are fairly economical to get and use, that meet most or all of a user's needs, except data storage requirements. Even if you don't need the central facility for computing, you probably will still require it for data storage. The storage devices that Professor Miller spoke of before, such as the IBM 1360 photo-digital store, are generally beyond the budget of any individual project. A million and a half dollars is a lot of money for a storage device, and it's almost beyond belief that any one individual project could get that kind of support from a funding agency, particularly these days. To get mass storage capability, a consortium of users will be needed, and with it will come a need to more carefully look at the problems of communicating between outlying computing facilities and the central file system.

We're getting started, but it's going to take a while. Looking at the problem primarily from the viewpoint of relatively low speed terminals, we were fairly complacent about the whole problem. The library project jolted us because their graphic display requirements demand a much higher data rate than we're equipped to deal with right now.

I'd like to go on to another problem: reliability. In a day to day practical sense, reliability is the central problem we face. When one talks about hardware, one of the things that's overlooked is the fact that the manufacturer tends to think about reliability in terms of availability, rather than incidence of failure. I think this is one reason why I don't really personally anticipate much improvement from the manufacturers, as far as reliability is concerned. Reliability is a different thing for the facility manager when he has users hanging out on the ends of terminals. We can much more tolerate - though maybe not so the library - four or five hours of being down, than we can constant interruptions that occur from intermittent hardware failures that drop the system. If every ten minutes the system dies, because of some minor glitch, the psychological impact on a user at a terminal is much more intolerable than if we simply tell him, "Well, we are going to be down four hours." He'll go off and play golf or something and come back and use the terminal. But if you keep dying, and the terminal has a tendency to go dead - that's a quite intolerable situation. This is one of the reasons we're installing the PDP-9. The PDP-9 was selected because it represents old technology, established equipment design, and has a good reputation for reliability. We at least hope to maintain terminal connection and let the user play tic-tac-toe while the main system is being repaired. We often overlook that problem when we design systems. We focus too much on total availability and not enough on incidence of failures. Incidence of failures is our biggest problem, not total availability.

Now I think I'll stop talking and let you ask questions; if you don't have any, I'll simulate some.

Stanford Computation Center
Campus Facility 360/67 Configuration
February, 1969

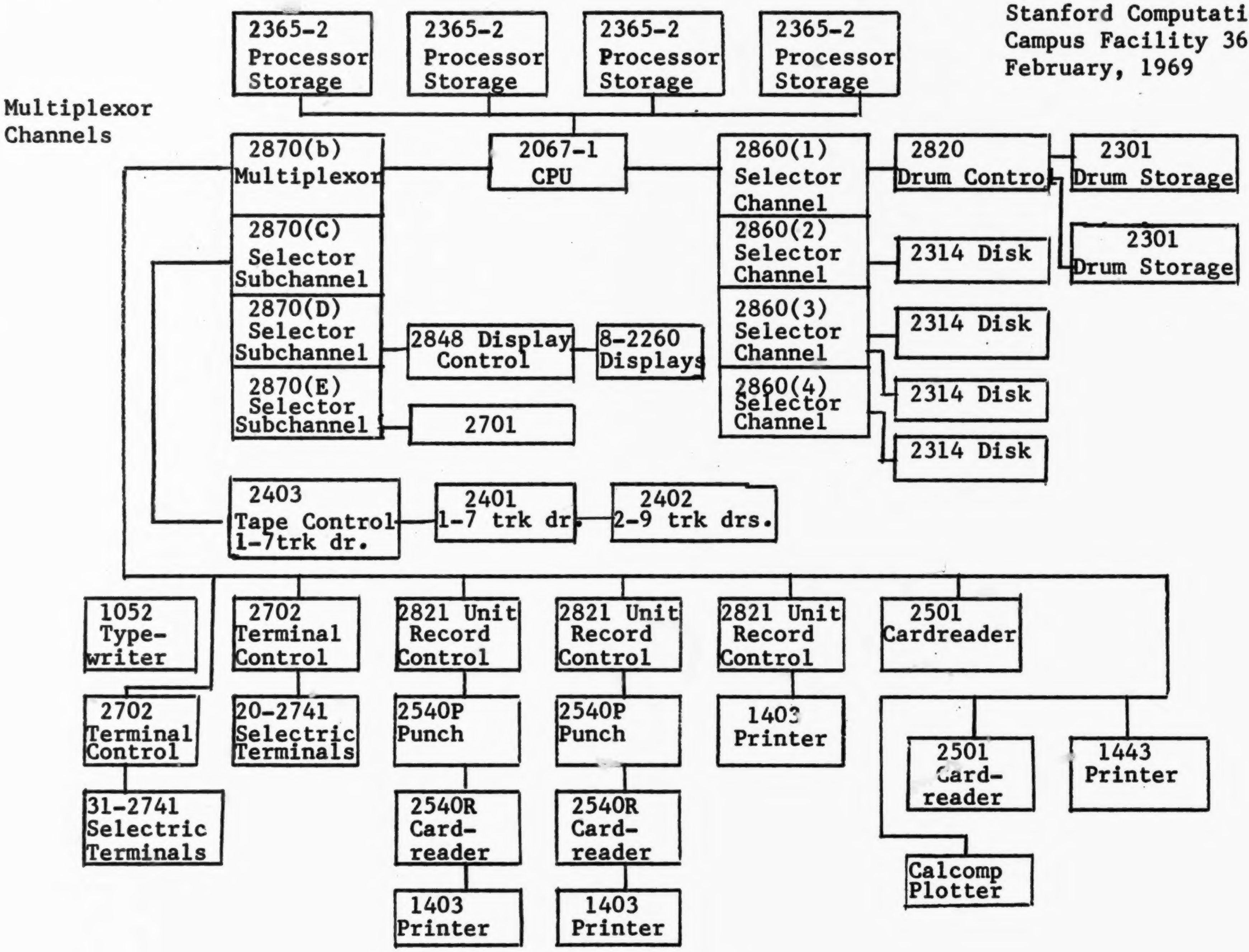
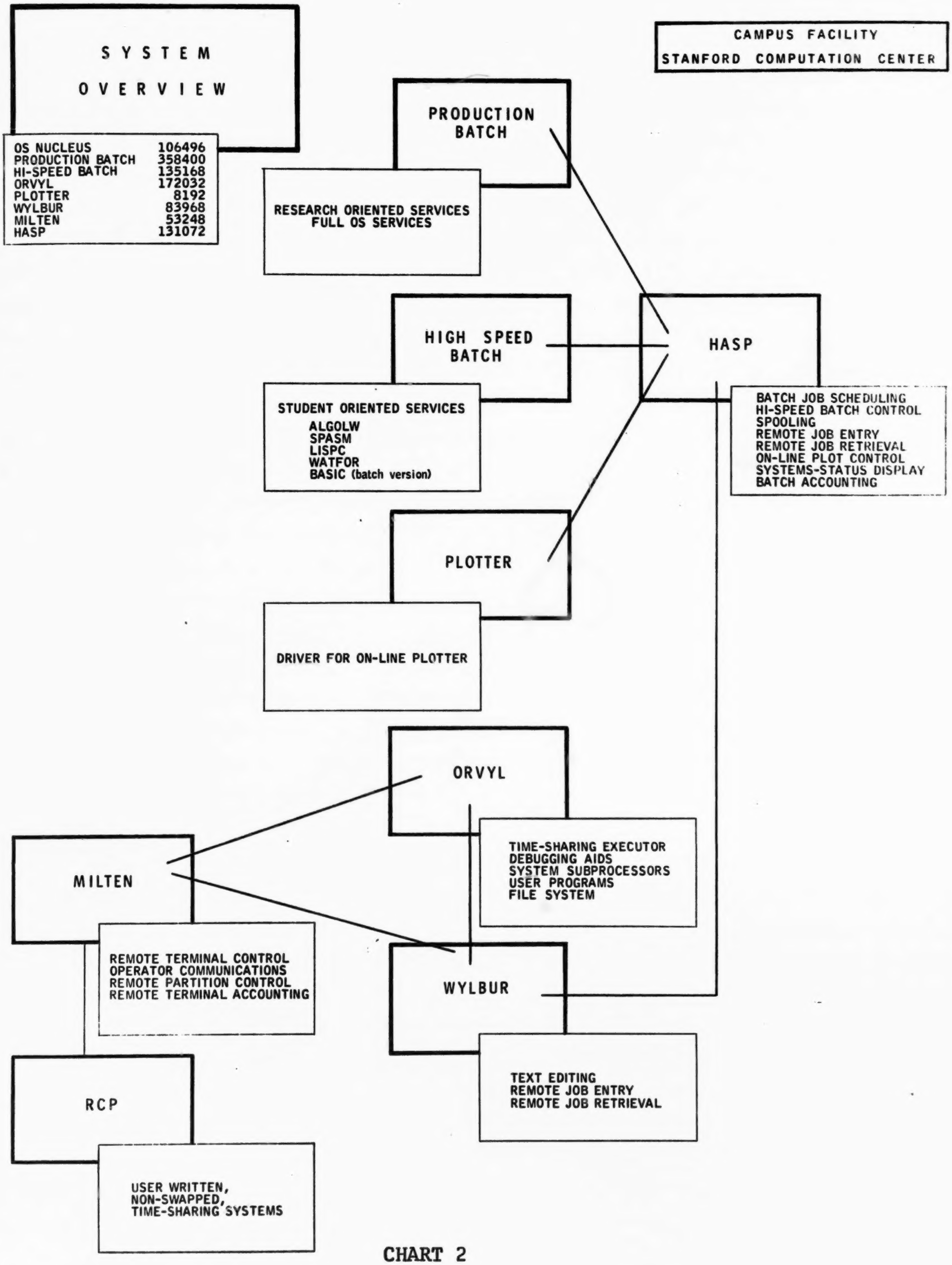

Multiplexor Channels
2365-2 Processor Storage
2365-2 Processor Storage
2365-2 Processor Storage
2365-2 Processor Storage
2870(b) Multiplexor
2067-1 CPU
2860(1) Selector Channel
2820 Drum Control
2301 Drum Storage
2301 Drum Storage
2870(C) Selector Subchannel
2860(2) Selector Channel
2314 Disk
2870(D) Selector Subchannel
2848 Display Control
8-2260 Displays
2860(3) Selector Channel
2314 Disk
2870(E) Selector Subchannel
2701
2860(4) Selector Channel
2314 Disk
2314 Disk
2403 Tape Control 1-7trk dr.
2401 1-7 trk dr.
2402 2-9 trk drs.
1052 Type-writer
2702 Terminal Control
2821 Unit Record Control
2821 Unit Record Control
2821 Unit Record Control
2501 Cardreader
2702 Terminal Control
20-2741 Selectric Terminals
2540P Punch
2540P Punch
1403 Printer
2501 Card-reader
1443 Printer
31-2741 Selectric Terminals
2540R Card-reader
2540R Card-reader
Calcomp Plotter
1403 Printer
1403 Printer

CHART 1


SYSTEM OVERVIEW
CAMPUS FACILITY
STANFORD COMPUTATION CENTER
PRODUCTION BATCH
RESEARCH ORIENTED SERVICES
FULL OS SERVICES
HIGH SPEED BATCH
STUDENT ORIENTED SERVICES
ALGOLW
SPASM
LISPC
WATFOR
BASIC (batch version)
HASP
BATCH JOB SCHEDULING
HI-SPEED BATCH CONTROL
SPOOLING
REMOTE JOB ENTRY
REMOTE JOB RETRIEVAL
ON-LINE PLOT CONTROL
SYSTEMS-STATUS DISPLAY
BATCH ACCOUNTING
PLOTTER
DRIVER FOR ON-LINE PLOTTER
ORVYL
TIME-SHARING EXECUTOR
DEBUGGING AIDS
SYSTEM SUBPROCESSORS
USER PROGRAMS
FILE SYSTEM
MILTEN
REMOTE TERMINAL CONTROL
OPERATOR COMMUNICATIONS
REMOTE PARTITION CONTROL
REMOTE TERMINAL ACCOUNTING
WYLBUR
TEXT EDITING
REMOTE JOB ENTRY
REMOTE JOB RETRIEVAL
RCP
USER WRITTEN,
NON-SWAPPED,
TIME-SHARING SYSTEMS

OS NUCLEUS	106496
PRODUCTION BATCH	358400
HI-SPEED BATCH	135168
ORVYL	172032
PLOTTER	8192
WYLBUR	83968
MILTEN	53248
HASP	131072

CHART 2

Discussion

Kilgour: Can you tell me more about the CRT you are considering? Is is manufactured by IBM?

Fredrickson: Mr. Lieberman, would you comment on that?

Lieberman: It is a disc storage device having straight alphanumeric and some graphic capability. The proposed system would be serviced by a controller located at the Campus Facility, with coaxial cable laid from the Campus Facility to the Library. One coaxial cable of video bandwidth would be laid for each remote terminal. As it becomes less efficient to lay a coaxial cable for each remote terminal, and more economical to group remote buffers, we would retreat from having twelve separate coaxial cables for twelve separate displays. We would probably use one coaxial cable, remote the buffer, and then drive several of the displays off one remote buffer. To clarify, right now all the hardware would be local to the Campus Facility except the displays themselves.

Fredrickson: Part of the problem is that displays would be going to another location. They're also going to the Institute for Communication Research, which is closer to the Campus Facility than is the Library.

Kilgour: Is it part of an experimental library system, or is it a system that is generally available and working?

Lieberman: The manufacturers have sold a couple other systems already and they're still expanding their capabilities. For instance, they expect to announce a keyboard entry system soon. Right now, though it's a passive display system.

Kilgour: Is this analogous to the IBM 1500 system?

Fredrickson: Yes, the basic technology is the same.

Kilgour: What takes the place of the 1800 CPU in the IBM 1500 system?

Fredrickson: Two things, First there is a controlling unit as part of the configuration; the rest will be the central CPU.

Kilgour: So it's central CPU driven?

Fredrickson: Yes, for the most part. The IBM 1500 system does other things in addition to driving the CRTs.

Veaner: The $500,000 figure you quoted for the network, is that just for external wiring to, say, a junction box in a building? That would not include any internal cost.

Fredrickson: This includes trenching and pulling, but probably not in-house wiring other than to a terminal box.

Shoffner: Would you review the two alternatives that you're considering with respect to how to tie the displays into your system.

Fredrickson: The system currently communicates between the brothers and the father --MILTEN, ORVYL, and WYLBUR. MILTEN is responsible for terminal communication, and it accomplishes this through two control blocks in the system. The first is a remote terminal control block (RTCB) and the second is a remote terminal buffer (RTB). MILTEN'S responsibility is to take the remote terminal block and the remote terminal control buffer and make an outside partition-- a "pseudo-remote terminal buffer"--which includes not all of the information that MILTEN requires, but only information that we feel that user application code would require. It then places this in a queue, which is basically just a collection of pointers, for the applications code to interrogate, and then pick its information off. When it wants to send information out to the terminal, the reverse occurs. The application code prepares this buffer asks for a supervisor service which hangs it on the queue, and then MILTEN takes it off the queue and sends it to the terminal. That's one way also to deal with the graphic display devices. That would be, I suppose, the preferred way. The problem is that the volume of information that is passed in CRT support is considerably more than with terminals like the IBM 2741 or teletype. The amount of time we spend spinning around in the commutators inside of MILTEN polling and in the other parts of the system is considerable, and to increase the data rate through that path probably would substantially impact the system's response time. This is why I'm a bit conservative about that approach and to tying it in, though it's the most natural way. It's simply an extension of our current system philosophy. We're already seeing an effect of it. We found in the time sharing system that we jumped the amount of time spent in MILTEN by almost four percent over what it was before we started time sharing, and part of the reason for this is that now, rather than going through MILTEN on a line by line basis in response to terminal interaction, we're driving through MILTEN under program control. That is, we've got code generating messages, and they generate messages too fast, and we're starting to swamp MILTEN a bit. That's the way we're currently operating and that's the way I think the Library's project would like to see graphic devices hooked on, but this is the reason I think we probably won't be able to do it for them. Not that it can't be done technically, but its effect on the system is going to be too hard. The other way is simply to make it look like a standard OS device to them. Just as they can now write OS files for their partition, and talk to any device we would provide some local support like we did for the 2260's to support them directly.

Shoffner: With the current configuration, you can get to any of your different services from any terminal?

Fredrickson: Yes.

Shoffner: So a terminal could sign on and get into SPIRES or into BASIC time share let's say.

Fredrickson: Yes.

Shoffner: If you change your organization, the library terminals will not have access to the non-library partition, and vice versa with respect to the other terminals. Is that a correct implication?

Fredrickson: Yes. Displays would only be driven by SPIRES/BALLOTS code, they would not be part of the more general system.

Shoffner: But you could write a little package in SPIRES that would hook you back through MILTEN to get to BASIC which is in ORVYL?

Fredrickson: Right. However, because they are going to load up the RTB's we don't want them to come in that way to begin with, an I'd probably have to prohibit it.

King: Well, if you provide a separate connection, for the graphic devices, does that mean that they would be time shared or not?

Fredrickson: They would not be time shared. If they are available to the library, they will probably not be available to the general time sharing services. After all the library's paying for them.

King: Within the library service itself, would it be time shared?

Fredrickson: To the degree that SPIRES is capable of dealing with multiple users, it would be time shared, but that's an application code problem, not a systems problem. I'm not trying to be rude, I'm just saying, that portrays a difference between me and the library, in the sense that that particular question is their problem from the systems point of view. They're not within the normal time sharing services. They're outside the normal time sharing services. They are, as a core resident system, a theoretically reentrant or reusable body of code that's capable of servicing multiple users of systems services.

Weisbrod: Did you mean that it was time shared or that it would handle more than one terminal simultaneously? There's a difference isn't there?

Fredrickson: I don't know; Ed Parker, how do you do it? Do you time slice or do you service to completion?

Parker: We service to completion in the sense of servicing until we get an I/O call, and then pass on to the next user. Our code is disciplined in such a way that no segment of code is too long. Then we pass on to the next user. In other words, we've got a very special purpose, "time sharing system" that doesn't have a clock associated with it, where we process disciplined segments of code to a logical stopping point, such as an I/O call, before we go on to the next user.

Fredrickson: This is the same technique used on the Real Time Facility's PL/1 compiler. They do not time slice that. They break it on natural units. ORVYL, on the other hand, is time sliced.

Parker: I might take this opportunity to say why we're not going under the general time sharing system. I think it would make Rod's life a lot simpler if we would. The main reason is that we don't want to lock ourselves into the dynamic relocation hardware of the 360/67, because then we would not have the compatability to switch off to some other hardware such as a 360/50, a 360/75, or whatever. We'd be locked into particular hardware, a particular non-OS system, and would have great difficulty in going on to a different machine. For the time being, we're staying with the standard OS in a way that allows us compatability through the 360 series without being locked into that particular hardware feature.

Fredrickson: I might make one comment. Here is a user, "a customer of a service," who's saying "I'm only going to buy your service just so long as I need to." It makes it difficult for us, as the provider of that service, to go too far in helping him. Now that sounds like a very brutal and rude thing to say publicly. In fact, we are not really that far apart. I point it out because it has a lot to do with how the facility views a customer, and how far it can go in trying to provide service. There is a danger in building up a system with all these "glitches" to support. A central computing service has a responsibility to try to keep some sort of stable load on its system at all times. At present, this project is pouring in funds and wanting to put code in here and tear it out there, but then, all of a sudden, say in one month period, the project is gone, and you've got operators standing around and all this equipment. You get into a lot of trouble.

Kilgour: I take it that really what's going on is that you're writing the library partition in PL/1 for a 360/65, so that it would be more widely useful in other communities outside of your present computer center. Is this what one of your aims--?

Fredrickson: I don't know if the latter is what motivates Ed's comment. I think the main thing that motivates is the feeling that as storage requirements go up, and as usage goes up, the capacity of this machine to meet both its original mission of providing general purpose computing to the university community, and also providing information retrieval and library services, will be exceeded. Now whether the goal is to export SPIRES/BALLOTS, I don't know. What motivates you, gentlemen?

Parker: Getting the job done. Let's do that first before we think about export. We have to have a product first.

Veaner: I would add that if we do take the direction of a separate machine as suggested, we would expect it to operate within the full context of the Stanford Computation Center. We do not expect to have a lot of operators and hardware hanging around doing nothing, but hopefully it would be technically and economically feasible to reassign them.

Fredrickson: Actually, I think things are going very well as far as the project is concerned. I got a little nervous the other day when the system died and it looked like it was caused by a SPIRES terminal. I don't think it was but they're sort of a scapegoat right now. [Laughter].

Reimers: Dr. Miller talked about a dichotomy between centralized computers and free-standing computers. But really, you're talking about a confederation, aren't you?

Fredrickson: I think that's the way it's going to end up being. I think it has to. As I said, we're spending 5.4 million dollars annually to support computing services. It's being used extremely inefficiently. I'm speaking now as a technician. I think it's being used inefficiently from a personnel point of view. SLAC has a 360/91 with a tuned system I bet they can't keep satuarted. Yet, I think there are technical ways of keeping that machine busy and economically justifiable. There is work on the 91 that doesn't belong on the machine. There's also work on the Medical School machine and the 67 that doesn't belong there either.

There's another problem that is not seriously being faced by the library project today, and that is the connection between what the project is doing right now and the problem of interfacing with the Controller's Office and budget and Administrative Data Processing. There's a complete vacuum there. I don't know how you can talk about a book acquisition system that's useful as an information retrieval system without closer liaison.

Parker: Equal time, please. The problem is not sitting in a complete vacuum. It's being worried about extremely hard. It just hasn't come to the stage where we've negotiated with you on the specific hardware or system to interface. It just hasn't developed to the stage where we're coming to you and saying, "Hey, Rod, we need this kind of interface."

Fussler: Are these machines owned outright? Or are they rented?

Fredrickson: The 67 is currently leased and that probably will be changed. The 91 is purchased. The 50 is leased.

Fussler: How is the cost of purchased machines built into the budget? Is it amortized?

Lieberman: Of the 5.4 million annual, about 1.1 is for amortization of owned equipment.

Rogers: Rod, I think you've stunned the non-technical people here, but we're certainly very much obliged to you.

Veaner: May I just take this opportunity to publicly thank Rod for his frankness and candor in pointing out the many technical and economic problems that many people, including ourselves, have been ignorant of.

Computer Operating Systems and Programming Languages: A Critical Review of Their Features and Limitations for Processing Bibliographic Text

Thomas K. Burgess
Manager, Systems Development
Washington State University
Pullman, Washington

A Paper Prepared for the Stanford Conference on Collaborative Library Systems Development, October 4-5, 1968

Rogers: Tom Burgess is going to present the next paper. He was born in Plattsburg, New York in 1933. He received his B.S. from Washington State, his Master's Degree from Stanford, and is presently a Ph.D. candidate at Washington State in the field of information science. He was on active duty with the Air Force from 1954 to 1965, serving as an Intelligence Officer from 1954 to 1960. He was a systems engineer at the Rome Air Development Center from 1963 to 1964 and a project scientist at the Office of Scientific Research from 1964 to 1965. He became a systems analyst at Washington State in 1966 and took over the management of systems development at Washington State this year.

* * * * * * * * * * *

I was asked to talk about operating systems, but it seems like we've spent a good deal of time with them this morning. I almost feel like starting over on something else. For the benefit of those of you who are not technically familiar with computing, it might be well if we view the operating systems again from the standpoint of the user and not from the standpoint of the computer scientist or the computer center directors. This morning Prof. Miller viewed the operating system from inside itself; that is a view of the operating system as the operating system sees itself. My view I think will be more turned toward the way the user sees it, and the way the user sees the obstacles that are caused by the operating system when he is trying to get his particular task accomplished.

First, we need to define an operating system: it is a collection of programs which provide for servicing of what is loosely called jobs or tasks, the things that you and I submit as programs to the computing center. We have operating systems because computing facilities are rather expensive and an institution must try to get the greatest amount of efficiency out of a system. The basic idea is to provide a job stream which most effectively uses the computing facilities. In the early days of computing one could get "hands on" the machine. One could sit down with his job and the computer and play with it. One could work his job all the way through or just portions of it. But demands on the equipment progressively increased and soon there wasn't enough time to allow everybody to schedule his own time on the system. The institution couldn't afford to buy more equipment and much of the user's time was obviously "sit and scratch your head" time, while the machine sat there and waited. Therefore, system designers began to build ways and methods of reducing idle time by developing executive systems, or, as they were originally called, monitor systems, which allowed a more efficient utilitation of the equipment.

Let's take a look at some of the parts which make up this collection of programs. Operating system components consist of many things nowadays. First, there's a job control language translator. This provides a specialized language for you to describe to the machine the job you want to do, and what parts of the equipment you need to use to get your task done. The operating system wants to know your needs because it has another part called the job scheduler, which tries to allocate available resources to those needs at the time most appropriate for that need. Originally, job schedulers ran just against the JCL cards which were removed from the decks. The machine operators obtained a listing of the jobs in the order in which they should be run, and the machine operators then put the decks in this order and then ran them. This didn't work very well because the operator had to stop every once in a while to run through the list and schedule more jobs. This was not a totally efficient use of the system, so designers began to add "spooling systems," which could store all jobs in a queue. This permitted the job scheduler to look at jobs waiting to be executed, jobs that could be deferred, new jobs that should be added, and the jobs that were finished and could be removed. Thus, the job scheduler can at any time assess the total resource requirements and optimally determine which job should be run next, based on the requirements that the user established in specifying what facilities he needed for his job.

This means that the system is not now scheduling the total machine, but is scheduling the components of the machine to do a given task. The system now has to have some way of knowing when specific components have finished their tasks; for this purpose, it needs an interrupt capability. This is a method of handling all of the inner machine communications; that is communications between those parts of the machine that tell each other what they're doing, so the machine knows what is going on. The interrupt system also includes ways of checking for errors and methods for handling program interrupts. It includes a series of programs which looks at what caused the interrupt and on the basis of that , and the current job mix in the system, and the current status of the entire machine, decides what to do next.

Another group of programs in the compiler sections of the operating system is input-output. This makes effective use of the peripheral equipment on the machine. The user no longer has control of the way data pass in and out of the computer.

There are two other parts of the operating system that I think are worth talking about. One is the program library. This is a group of very frequently used programs that are stored in the machine. They may include nothing but small sub-routines or they may be very complex programs. Because this series of programs is used by many people, it's more effective to store them in the machine than to have them read in each time they are needed.

The last part of the operating system is the compilers and assembly languages with which the applications people to their work. They are also in this program library, as are most of the other program groups I have mentioned previously. I want to spend some time on languages, because they have many effects upon how we can do our job.

There are numerous kinds of programming languages around. It would be almost impossible to name all languages that exist. The earliest languages were assembler languages, which by original definition were one for one transformations from some language which was more easily understood by people, to the binary language that the machine understands, These developed into more complex languages that no longer really represent a one for one transformation. What is known as a macro instruction has been added. Macro instructions are small pieces of code that in reality are sub-routines, but which extend and add more capabilities to assembly languages than were available previously.

Another grouping is the so-called higher level languages. These languages allow us to communicate our ideas to each other more easily and allow us to program more easily. These languages represent a "one for many" transformation, i.e., one statement in the higher level language generates many machine code instructions. High level languages have had a diversified development. They tended towards specialization in accordance with activities and interests of their users, because users tend to develop a certain technical language with which they communicate with their peers. We now have a large number of higher level languages, each one devoted to specialized tasks. There are languages for civil engineers, architects, just about any kind of specialty. There is even serious talk about languages for librarians. Many of these languages are not frequently used. Many are not even always available for use on a particular equipment.

There are three major groupings of languages. First there are the algorithmic languages; they are the languages for the mathematician or scientist who wants to do complex calculations. Foremost among them is Fortran, a language primarily designed for those who have very little input data but who require a large amount of calculation with very little data to output. Hence, Fortran's input-output facilities are small rigid, and not very flexible.

Secondly, there are the business oriented languages, which were more or less thrust upon the industry by the federal government. These languages are designed for handling large amounts of numeric data with very little calculation involved-a little adding, subtracting, keeping track of business accounts, payroll, etc.

The last specialized grouping of languages, and it's difficult to pinpoint the most popular of these, is the list processing languages. These were developed by researchers working in machine translation; they needed capabilities for string manipulation, that is, manipulation of strings of alphabetic characters, which is what they were trying to do in machine translation. It's hard to pick out one of the foremost of these; SNOBOL is probably the most common.

Only very recently has there been any reverse in the trend of specialized languages to bring us back to more generalized languages. In IBM, the thrust towards a language called PL/1 is probably the only really good move in this direction. PL/1 is a relatively new language, and although its specifications are very clear, its implementation is somewhat limited. There's a big difference between a specification of a language and its implementation. PL/1 is getting better each time we get a new version of the operating system; the language is much improved, better defined, and the compilers are much more efficient at producing smaller amounts of code which run faster on the machine. In PL/1 we have a combination of qualities: the algorithmic capabilities of Fortran, the input-output capabilities of Cobol, and the string manipulation abilities of the list processing languages. In PL/1, it looks like we are getting the type of language which is at last capable of meeting most of the requirements for library applications.

But again I say, there is a significant difference between the implementation and the specification of languages, and it depends upon the particular computer. This is the reason why you'll find that in some cases a program compiled in one language at one location will not run on a machine in another location which has the same kind of compiler. The impact of these languages on program development means that we really have to look at the job we want to accomplish, and once we've figured out what that is, then we need to pick the language in which we should write. This doesn't mean that we can say that for the total library automation task we ought to use PL/1 for everything. This means we need to look at each of the individual tasks. There are many things that can and should be done very effectively in assembly language; many can and should be done in Cobol and Fortran as well. We have to look at the task.

Another reason for using these higher level languages is ease in programming maintenance. As the operating systems change and as our requirements in the library change, we find that it's necessary to modify existing programs. If you have a program that's specified in something that looks like English, it's easier for somebody who never saw it before to understand it. And so it's better for us to write in the higher level languages because of this ease in programming maintenance.

Now let's go back to operating systems, and look at the criteria behind development of operating systems. First of all, the main purpose of an operating system is to maximize component utilization i.e., the CPU, all the input-output devices, all the storage devices, and all other units.

4

The second major function is to provide better user services. This means that the people who design operating systems look to see if they have a large number of jobs in a queue waiting to be processed. To the designers, each job represents a user. Here we have a "one user, one job" idea on the part of these system people, so that they treat each job with equal priority in terms of trying to meet user requirements. We all know that this is not true; for instance, payroll jobs are not "one user" oriented. There are many other multiple user jobs and certainly the things the library wants to do represent many users, not just a single user. A second bias in user services is the "short job bias." It is a direct consequence of the "one job, one user" bias. In other words, if we can run a whole lot of short jobs through the machine, then we've satisfied more users. We all know that in most cases we don't have short jobs in the library. The last bias is against jobs that require a large amount of input and output. Again, this is based on the requirements of users that have short jobs with little I/O. But library jobs use a large amount of input and output time because they tend to be involved with massive strings of characters.

The third criterion in designing systems is ease of software maintenance. Systems are dynamic and the computing center's requirements are dynamic; system configurations must change, so the system should be designed to make it easy for the system programmer to get in and maintain it. It should also be designed so that he can easily extend the system to cover the new equipment.

Now I want to outline a few of the major operations problems in university computing centers.

The first problem is the wide job mix which the center must perform. It is faced with extremes of complexity that one does not find either in a service bureau or specialized, single purpose facility. The job mix ranges from the kinds of programs that physicists and chemists run for 16 hours, to the student program which takes longer to load in the machine than it does to process. Fundamentally, operating systems were not designed to cover this tremendous job mix; they were designed to cover the job mix that is found in most service bureaus or in a single research or data processing center. To cover this wide mix, users are sometimes forced to make modifications to the operating system.

A second problem, one of a political nature, is scheduling: do we schedule jobs automatically on some equitable scheduling basis, or do we establish a priority system? This can produce quite a severe political hassling between the computer center and their users. The center would prefer to do it on a completely automatic scheduling basis, but they haven't been able to achieve this goal. But as soon as you allow any kind of priority, then everybody wants a priority. On college campuses the computing centers are usually tied fairly closely to the Computer Science Department, and this can constitute still another problem. The Computer Science Department treats the computer as its own piece of laboratory equipment; it is theirs and for their use alone. The Computer Science faculty and their students take this possessive attitude which conflicts with the rest of the users and those who are running the system. Computer Science people come in and want to have their job put first in the queue. Well, if the operators are students and are taking courses from the faculty members, they probably will get their job placed first in the queue. So again there is a priority problem.

Lastly, university computing centers face financial constraints, both in terms of support for maintaining the system itself, i.e., in providing an adequate systems staff to meet all of the university's requirements, and in providing adequate equipment itself.

These are some of the problems. How do they affect us in the library? First of all, as most all of you know, library jobs are input-output bound, not only from inside the machine, which is moving data from disk storage into the CPU for processing, back out to some other storage device, but also in processing from an input medium into the machine, and then out into some output medium. All of our jobs run up against the short job bias. The bias looks like the super-market express line; if you have less than 10 items, you go to the express counter and get serviced right away. Except that in a computer, you've only got one counter and with your big basket or two big baskets, you have all these little people with less than ten items popping in front of you, and if there's enough of them, you're not at the head of the line anymore, you're at the end. With that kind of problem, how do you get to the head of the line? The only way is by some intervention in the operating system that provides you with some internal priority in the machine which says, "No more people are going to be placed in front of this job; it's going to be done." This usually means some manual intervention by the computing center staff. At this time, there isn't any way of automatically looking at a clock and saying, "This job has been in for eighteen hours, we had better get it to the top of the queue."

Spooling has provided a whole series of new problems which we never thought existed before. Many of us have grown up from the punch card era. In punch card jobs, we tended to build little programs and link them together into a stream of programs which we wanted to run sequentially. In those days--and we still design things that way--after successful completion of one job, we wish to run the next job. Now we have spooling, not only on input, but also on output. With spooling, a job is run in the CPU and a data set built in some external file. At some later time, again according to priorities, it is printed or punched or returned to your terminal. With big operating systems being not too stable in operation these days, there can sometimes be troublesome problems between the time the job gets completed and time you get the data printed out, and sometimes you may never get the data printed out, and it's lost entirely. But as far as the machine is concerned and as far as any of the programs that you wrote are concerned, that job ran to a successful completion. So your next job is going to be run whether or not you've gotten the first job out.

We also found in our computing center that a job was a job, i.e., it was treated completely independent of anything else you might want to do. If a job bombed out in the last five seconds of operating time, for instance, the normal procedure was to put it back in the stream and do it again. If you're talking about overdue notices from your circulation file, and you set a status bit that says, "Yes, I've now printed an overdue notice," and you ran that job again, you're not going to get any notices printed, because they've already been produced according to the file. What we lack here is inner communications from the output spooling queue back into the program, so that you can say, "Yes, indeed, now that I have printed output, I have truly completed the job." Only then can you go on to the next job.

Without this communication, a different kind of program design is required. Now we actually have to provide a physical time lag so that we can get the printout in our hands before we submit the next job into the queue. We can't submit them all together and hope that the system will run. What this has done is lengthen our turnaround time; many jobs that we originally expected would run overnight now take two or three days, because we have to wait to get actual outputs in hand before saying, "Let's go on."

It was a rather rude shock to many of us, that you can't go to the computing center and say, "Look, I've now got some money and I want to hang a bunch of devices on the machine to do a new job. With the operating systems that we have today, it just isn't done. Things have to be coordinated. Devices can be physically hung on the machine, but they have to be supported with software packages, and in many cases we have to write these software packages, and they are complex and take a long time to write. Also, if we are going to add additional requirements and facilities to computing centers, we've got to give them some lead time.

We talked this morning about "fail soft" and degradation of systems. The operating systems that we have today provide some of this capability. We need more of these capabilities; we can't, for instance, properly manage our personnel in the library if we can't guarantee that we can get at least some part of our machine processing done each day. It is difficult to find jobs for your marking section when you don't get book labels or pockets from the computing center. If this happens often, pretty soon it's difficult to figure out what you're going to do with all of these people. You can't send them on an eight hour coffee break! We need to recognize that systems are unreliable and go down for many reasons. We must try to build into our system, either in our own application designs or in the basic design, an ability to degrade our activities and still get something done.

And then we must always realize that we are going to have catastrophic failures; power failures are the most notable of this kind. If we're partially through a lengthy job, it's uneconomical to go back and re-do the whole job; we should try to pick up from some point and go on forward. This is known as "check point restart," and it means building in certain plateau levels in the processing of the program which--if you fail, you need only fall back to that last plateau, and go on from there. This was brought home to me very strongly when I was building intelligence systems for the Air Force. We were nine hours into a ten hour sort when the power failed, and we had no check point restart. We had to re-do those nine hours.

We've got to insist on better reliability within our total system. We now talk about building systems that are real time, on-line, and yet these systems are of no use unless we can insure that they are working all the time. When you go to a real time system, you can't fall back to a batch system. In many real time systems, it's all or nothing. So you've got to build in reliability. All of these things cost money, and you have to play one side against the other, until you've reached an optimum solution. You must decide how much reliability you can afford or how much you can not afford.

Another implication concerns maintenance. On third generation systems, system maintenance is not transparent to the application program. Systems keep changing. Stanford is on version 13 of the operating system; we're on version 14 of the operating system, and we have 15 and 16 in hand, and on and on it goes. In the year 2000 we'll probably have version 979 of the operating system available to us. Because many of these changes are not transparent to applications programs, all of a sudden the programs which worked beautifully for three months are now in terrible shape, and you don't know why. Well, you find out shortly that the trouble is due to changed operating systems. Now you have to perform some maintenance to make them work with the new system.

What are some of the solutions to these problems? One of the first solutions that pops into most people's minds in fighting the scheduling problem is to get his own computer. You can pat it, and if it's working it can run your job when you want. As you can gather from the above discussion of operating systems, if you're going to have your own machine, then you've created for yourself the same basic problems the computing center has. You had better be prepared to face this possibility, and it's expensive to provide adequate expertise in terms of system programmers to maintain the system. It's not the same as installing a 407; it's an entirely different kind of ball game.

What we really need to do is sit down with the people responsible for computing activities on the campus, and with them design a total computer system which is adequate for all campus needs, and buy a system of computers. I don't necessarily mean just a single CPU, but a complex arrangement of computing power on a campus which will meet all of the requirements of reliability, fail safe, fail soft, check point restart, etc. We need these capabilities or we can't live. We can't live in an environment unless we have redundancy or flexibility in the system.

We talked this morning about system redundancy, i.e., a second unit or copy of the first. But this isn't always necessary. How about flexibility in the system? By building in certain kinds of compatability between different machines, a job which normally runs on one machine, if that machine is down, can be run on another one. Building in a degree of flexibility allows for degradation of the total system. A smaller machine will take a little longer to get a job done, but at least you're getting something done. We must recognize that there are weaknesses in operating systems, so that you can compensate for the problems.

As we move into the world of on-line, real time operating systems, we must be able to recreate information in case of a catastrophic failure; adequate systems will allow you to recreate this information. This is "backup," and one also needs "backup for the backup," because there are times when you're copying data sets on tape so that you can store them away for just that kind of eventuality, and that's the moment when you lose everything, and you've lost both your backup and your original file. Then you know you're in real trouble. So you've got to include in your design some "backup for the backup." I should conclude by emphasizing that this is the most important thing you can do in designing a system.

Discussion

Weisbrod: You mentioned check pointing. The more complicated the system under which you run, the more complicated it is to design any kind of check point facility, because you have less of the machine under your own actual control.

Burgess: That's right. And that's the problem in trying to build a check point restart capability in spooling systems that allow you to go back and start over again. You see, you've really lost control of the machine. All you can do is specify some things that you'd like to have done and the machine decides when or if it is going to do them.

Weisbrod: I was wondering if I could direct a question to the people at the Stanford Computation Center: what kind of check point facility do you have?

Fredrickson: Well, there are check point facilities for the user disc; they're the biggest problem you experience under the IBM operating system. OS in its virgin state, is particularly capable of wiping out volume directories. In the beginning we check pointed nightly all the discs in the system, just on the off chance that the following day a volume would be lost. It got so it occupied about four hours a day, and we couldn't afford to do it any more. We modified OS so that it won't clobber discs anymore, no matter what happens, and we have since discontinued the policy of check pointing discs today. We have a public statement policy which is only protective of the Computation Center. (Laughter) We kept track of data sets to see whether they were changed or not or just used (which OS, of course, doesn't do), and we were able to determine that less than ten percent of the files in the system were changed daily anyway. And yet, until we found this out, we had no way at the time of check pointing anything other than whole volumes. It would be less costly now to go back and start check pointing changed data sets, but now that we've got the users trained to protect themselves, I'm not so sure we want to go back and assume the responsibility.

Kilgour: What in your opinion is the most effective way for the user to protect himself?

Fredrickson: Well, much depends upon the application. That's the foolishness of expecting the central facility to assume responsibility for protection. We don't often know what is critical and what isn't critical. In general, we try to show people how to prepare programs that involve large volumes of data in such a way that they can restart their programs without having to start from the beginning. We recommend that people who have put in many hours in preparation of data files tape them at that point in their operation where it would be costly for them if they lost their data, *but they're the only ones who can make that judgment*. We don't know how many man hours they're put in preparing that volume.

Kilgour: Of course, the applications program could destroy it, maybe more readily than your operating system.

Fredrickson: Well, an error on the part of a clerk putting in *delete* rather than *keep* on a JCL card is certainly going to be devastating.

Kilgour: You don't have any routine recommendation then. If it's a permanent file, copy it onto a tape.

Fredrickson: Actually, we do check point, but we don't tell the users. It's not so much to protect them, but to protect our income. If all of the data sets disappeared off the system, it would take many months for the users to get back to spending the money at the rate we require them to expend.

Burgess: One of the criteria we use in designing online systems, is that if you're keyboarding information, to recreate it, you would have to re-keyboard it, and it's best at the time you are modifying your online file, to duplicate those records you're changing at that time. Each record that you change, you should write someplace else, where you can get at it.

Kilgour: Where is that someplace else?

Burgess: Well, in our case, where we're using a data cell, when we write records on the data cell, we store a copy of each changed record on disc, and then later copy that disc on tape.

Fredrickson: In the Stanford system we try to protect the user from system failures that might hurt his data, and for the most part we've been successful in recapturing information that had been on direct access storage even after OS has not been able to recapture it. We've written a number of special programs to go in and untangle things. If the user destroys himself, this is something we really can't handle, because he can do it in so many different ways, and we can't really prevent it.

King: I would say that that's the experience at Columbia also. It isn't the system that often clobbers people, it's the fact the user doesn't understand some feature of the system, and a most common thing (I've just discovered a number of instances of this to my horror) is that people will update some direct access file and introduce a lot of transactions. They'll be updating this file and in the midst of this activity, one of the terminals will cause the whole system to crash. Then the system automatically restarts at the beginning of the job that was currently on, which means it starts updating the file from the beginning of these transactions, introducing duplications. That's the way the system works, and everyone is presumably informed that that's the way the system works, but there are people who design production programs with the expectation that the unexpected will never happen.

Fredrickson: And they're wrong.

Burgess: If you're running multiple jobs at the same time, operating systems are supposed to protect all other jobs from mistakes that one job can make. In reality, the operating systems as they look today cannot anticipate all the mistakes that application programs can make, and so they haven't been able to field all these problems. When this happens, the operating system gets confused, and pretty soon everybody is wiped out. University students are very good at finding out new ways to clobber the operating system.

King: Most of the time I think that you can figure out how to prevent people from crippling themselves. But if you did implement all of these protective schemes, the cost of an individual unit would be higher than anything you would want to pay. The cost, for example, of simply duplicating in a disc and writing out on a tape every changed record on a data cell is horrendous. It's a lot easier to tell the user, "Look, we'll dump the data cell once a week, and if you want to preserve your system more often than that we'll provide mechanisms for doing it and lots of luck."

Kilgour: Aren't you saying "users" though? If it were one user out of your thousand, it wouldn't be a problem right?

King: It's true that the center has 2000 users.

Kilgour: The problem is users, not one user.

Fredrickson: You can write programs in such a way that they're less sensitive to system failure. Our experience with system failure is probably as extensive as anybody else's. We don't really have that many when you get right down to it. We run rashes when we install new equipment, but most of the time we're relatively stable, and the damage to users is relatively infrequent. We have to recognize that the loss of one user is not very critical to us. (Laughter). Most of the time we're quite concerned and that isn't when the user gets lost. Our main concern is viability of the system to the bulk of the users, and generally speaking, that's the way we bias operations.

Shoffner: What sort of crash rate do you have?

Fredrickson: Well, typically we don't die when we're running well and when we haven't just added new software. But the last two days in which we have been using the new time sharing system, we died three times the first day and twice yesterday. I don't expect to die at all starting next week.

Unidentified Voice: Do you have IPL's on that?

Fredrickson: That includes any IPL's. We IPL once in the morning and that's it.

Gwynn: Early in your comments you said that it was advised that each person contemplating developing a system attempt to associate with the large computing facility. Now we've seen contrasting views here. Weighed against the advantages of central services is the fact that most of the knotty problems that we're going to encounter in library programs have to be done by our own methods, and that is also the view of the relatively impartial view of the Computation Center, in the sense that they're trying to serve the masses rather than the individual, tailored, highly efficient production. So what guidelines does a person use? How come it's so obvious that we should associate with the large computing facility?

Burgess: Well, I really was saying that the operating system as it exists today in the large center does not provide all the facilities we would like to have. What we have to do is convince people of what we need, and have them do it, because it's too expensive for us to do it ourselves. You see, if we buy a machine, we're stuck with an operating system, and we're going to have to modify it to fulfill our special requirements. It's better to get a larger group of experts to help make modifications. It's not impossible to modify and design an operating system which does meet our requirements, but I think it's better to associate in this way and design an operating system which will meet your requirements as well as everybody else's.

Kilgour: When you say "ourselves" though, you mean one institution, don't you?

Burgess: No, I mean the library itself. I don't feel that the library itself should try to build up a staff of systems programmers to support-----

Kilgour: You're talking about a library in one institution? You're not talking about a group of libraries and a group of institutions.

Burgess: No, no, just one library.

Kilgour: For instance, in this collaborative effort, would it be worthwhile to develop an operating system?

Burgess: Economics will dictate this. Even if you are going to do it yourself, you better be prepared to pay the cost.

Fussler: I would like to comment briefly on the desirability of trying to maintain a standard operating system, on the assumption that this permits institutional exchange of certain kinds of software, as against local, developmental changes in the machine operating system, to make the local operations more efficient. Is there any virtue in holding back and in using a standard operating system that you know can be improved?

Burgess: No, I don't really think so, because generally what happens is that for your particular mix of work, the standard operating system is not that efficient for you. Your facility can easily get saturated and you have to do something. You either have to get more equipment to get the jobs done, or else make the modifications. In our case, the modifications were forced on us, because we couldn't handle the student job load.

Fussler: Well then, the consequence of this is modular elements in your application software probably couldn't be run on another 67?

Burgess: If they are dependent on specialized portions of the operating system, this is true in many cases. A good example is our acquisitions system, which is operating against a specialized terminal handling package which is different from Stanford's. We could not interchange those systems.

Fussler: We've tried to persuade the staff in our computation center that this is a problem that ought to be taken seriously in relation to the subsequent availability of applications software. But there's a good deal of pressure within the computation center to improve the quality of the operating system, and while they hope it may be "upward compatible" I'm less sanguine that in fact it will be.

Kilgour: I would guess, Herman, that the only way compatibility could be assured would be to have dedicated machines within the same operating systems.

Fussler: Let's assume that every university's computation center is changing uniquely the operating systems that are being used. Then the application software doesn't really become interchangeable in these machines. The principles will be, and obviously, one can recode.

Weisbrod:	A standardized operating system is also in the computer center's own interest, because the manufacturer's subsequent changes become more of a problem to the computer center which has its own locally tailored system. That's what Professor Miller was talking about.
Kilgour:	An operating system can have a bug in it for a long time without its being picked up, until some large and complicated application program, like the library system, comes along and finds that bug. Getting these straightened out locally causes real trouble.
Burgess:	It's even worse if that bug only shows up sporadically.
Kilgour:	That's the way it always is.

Developing a Campus Information

Retrieval System

Professor Edwin B. Parker

Associate Professor

Department of Communication Research

Stanford University

A paper Prepared for the Stanford Conference on Collaborative
Library System Development, October 4-5, 1968.

Introduction

My purpose in the short time available today is to attempt a general overview and a brief progress report on the development efforts of Project SPIRES, which is financed primarily by the National Science Foundation. At this stage in our development the name behind the acronym, Stanford Physics Information REtrieval System, is no longer quite appropriate, although we still hope it carries the connotation of high aspiration. Our close collaboration with the Automation Division of the Stanford Library and our commitment to provide the computer software for Stanford Library Automation Project has broadened our perspective and our goals. Funds from the U.S. Office of Education to Project BALLOTS are making it possible to take on that added responsibility. Because of this expansion of both systems and applications programming effort we are proposing a change in the name of our project to Stanford Public Information REtrieval System, as Allen Veaner mentioned yesterday.

Project Goals

SPIRES has two major goals. One is to provide improved information services to members of the Stanford community, beginning with the physicists who are serving as the first test population for our development efforts. The motivation is to take advantage of the new computer technology to extend information services to scientists and other users to a level unthinkable if it had to be provided by current manual systems. In our view, the main advantage of time-shared computing for information services is that, for the first time, we can build systems with two-way communication permitting rapid negotiation between user and system. In technical jargon we call these negotiations 'feedback loops.' It is primarily because of this two-way communication and the facility with which user interactions with the computer system can be recorded and tabulated in the computer that we are optimistic that the developing system will remain responsive to real needs of users. Meanwhile, during the development stage, we need the help of other kinds of feedback, including interviews and user tests of partially developed systems. The first level of new service that SPIRES will provide is what MIT's Project INTREX is calling the augmented catalog. For Stanford physicists this means providing the capability to search several document collections, including the Stanford Linear Accelerator Center preprint collection, Nuclear Science Abstracts, the DESY index of high-energy physics documents, and a collection of physics journal articles. Unlike the current library catalog there is an entry for each abstract in NSA, for example, and indexing under each major word in the title of each article, not just the first. Some collections, including the SLAC preprints, are indexed by footnote citations permitting searches forward in time as well as backward from a given article or bibliography. Since the goal is to meet the information needs of the users, it seems quite likely that there will be motivation to go beyond the provision of augmented catalog services to text retrieval and various forms of data retrieval as rapidly as the technology and available funding permit.

The second major goal of SPIRES is to provide the long run economic benefits of more efficient internal processing of bibliographic information in the library. In other words, the goal is to meet the computer software needs of Project BALLOTS as they develop their acquisitions, cataloging, and circulation systems. This goal is completely compatible with the first, for both technical and economic reasons. Technically, it doesn't make any difference to the computer programs whether the user is a librarian searching through a Library of Congress document collection, or the library's own In Process collection, or whether the user is a physicist searching Nuclear Science Abstracts, or possibly a small private collection of documents. In both cases the computer system permits the user to perform quickly what might otherwise be a tedious manual search. There are differences in the kind of output formats the different users will require--the physicist may want an alphabetized bibliography while the library clerk may want a purchase order produced. But these differences in output format are small variations in the application of a general system which must be the same for the major function, namely the retrieval process.

Economically, it may be necessary to meet this goal of improved internal library processing in order to be able to afford the improved information services to users that is the first goal of Project SPIRES. As we and our funding agencies have been finding out, development of time-shared computer systems is an expensive proposition. It would be difficult to justify the costs of such development on the basis of the improved service to a small number of users, even such reputedly affluent users as high-energy physicists. My personal suspicion is that once such systems are readily available and have proven themselves valuable, users will be quite willing to spend some of their research funds on the kinds of information service we plan to provide. Meanwhile, few people have budget items for an as yet unproven service. Looked at economically, the library's view that the internal processing service should be provided first, with the user services added as a by-product service, may be the correct one.

Consequently, we are able to collaborate easily with the library in the development of a system that will meet both goals.

Basic Choices

In attempting to provide expanded services to users and improved internal library processing, there are several choices to be made. The first choice is obviously whether or not to go to a computer system, and, if so, whether to go to a batch processing or time-shared system. The decision to go to a computer system does not imply replacing the present manual system. It means adding a searching capability that will permit librarians, library clerks, and scientific users to locate bibliographic information quickly and without drudgery. It will mean that bibliographic information, once typed into computer readable form, either locally or by the Library of Congress, will not have to be typed over and over again. It will mean that more than one person can look at the same part of the same file at the same time without getting in another's way.

This is not a matter of replacing an old manual system with a new automated system--it is a matter of giving the present personnel better bibliographic tools with which to perform their present tasks and freeing them from much drudgery so that they can take on new responsibilities and services. The computer will not be a panacea. It is not likely in the immediate future to provide full text service, for example, because of the high costs of computer storage and the high costs of keyboarding information into machine readable form. Consequently, we have to think in terms of providing visual display terminals that can display information stored on microfiche as well as information stored in digital form for computer processing. In short, the choice of using a computer system is not an either-or choice; rather it is a decision to add one more bibliographic tool to the equipment of librarians.

Stanford's choice of an interactive computer system permitting immediate response to queries rather than a batch processing system providing output at scheduled intervals was a choice that few libraries can take. If both kinds of computer systems were easy to provide, then I'm sure librarians would all opt for the interactive system. Stanford's decision was that a batch processing system, with the slow feedback associated with waiting for the next batch of computer output, would be unacceptable. At least the present manual system is interactive and operates in 'real time.' So Stanford's decision was to wait until interactive processing appeared feasible. However, there are perhaps less than half a dozen universities in the world (Stanford and MIT are the first two that come to mind) where the quality of computing and research in computer science make it possible to make such a choice today.

The problem is that most existing computer time-sharing systems are devoted to applications that do not have the massive storage requirements of large library and information systems. None of the proposed large scale general purpose time-sharing systems have yet been successful. For us to tread where IBM and others have so far failed would be foolhardy. We may be foolhardy anyway, but we chose not to wait for someone else to develop a general-purpose time-sharing system. Instead we forged ahead and are attempting a special-purpose super-simplified time-sharing system of our own. If we succeed, we are heroes, and if we fail we are merely visionaries who were ahead of our time. We are convinced that the right way to go for the long run is the interactive system and we are confident enough to think that our chances of producing such a system are good. But we don't recommend it for others unless they are confident they can work at the present frontier of computer systems development. This is not the same thing as writing applications programs for a well-developed stable computer system.

Most of the other major choices are choices of scope. Should we think in terms of a purely local information system or as a component in a developing national or international information system? Should we restrict ourselves to a single discipline, such as physics, or should we expand to include chemistry, medicine, engineering, social sciences, and humanities, etc.? Should we restrict ourselves to bibliographic information or should we expand to include management information such as accounting, inventory, personnel files, etc.? In attempting to meet the information needs of the scientists should we stop at bibliographic information or should we expand to full text retrieval (not necessarily from computer files), and to retrieval from large archives of non-bibliographic data? Will the computer terminals necessary for access to the retrieval system be special for the one application, or should they be the same terminals scientists have in their labs and offices for computer applications other than information retrieval?

I'm not certain that we're making the right choice at all of these choice points, but I can report briefly what choices we have made or are making. One obvious factor in making the decision is economic support. It's one thing to dream grandiose dreams and another thing to propose economically realistic projects during a time of budget cuts and cost-effectiveness evaluation criteria. Another factor is that we must avoid attempting something that is too complex to be successfully brought to fruition given the current state of the computer art. There are obviously economies of scale to be accomplished by making a system general enough to provide more than one kind of service (e.g. both a bibliographic and a management information system). At the same time there are complexities of scale as more general systems are attempted. The lesson of IBM's Time Sharing System (TSS) should warn us away from attempting too much if we hope to complete development within the time and budget planned. We don't want the complexities of scale to overwhelm the economies of scale. The special purpose systems are usually more efficient for the purpose they were intended to serve.

With these considerations in mind we are attempting first a bibliographic information system that is intended to be a local system that can serve as a 'retailer' outlet for the 'wholesale' products of the developing national information systems in the various scientific disciplines. We presume that although batch processing systems (like MEDLARS) may be more efficient as a centralized system, interactive systems will have to be decentralized to avoid the expensive communication costs. This judgment may change, of course, if there is a drastic revision of domestic telephone tariffs after the introduction of domestic communication satellites. Nevertheless, our best guess now is that there will always be need for local or at least regional service, even though there may be network switching to a national information center or centers for infrequently used material. Local systems should be more responsive to local needs than any centralized national system can hope to be.

We are assuming that few users will be able to afford computer terminals solely for the purpose of bibliographic searches, and that we must make our service available from whatever terminals users have. (At Stanford there are already about 100 typewriter terminals in use for remote computation and other computer services.) Expansion beyond physics references and beyond the collections necessary for the library's acquisitions and cataloging functions should be rapid as the appropriate machine-readable data collections become available, provided that there is a user demand and a means of financing the expensive storage costs. Some additional programming is necessary to translate each new data file into our standard internal formats, but that investment is small relative to the programming required for the retrieval system itself. Our prototype system, which we hope to have operational by January 1, was designed entirely as a bibliographic retrieval system, although it doesn't make much difference to the system whether the records being retrieved are records of books or whether they are personnel records. SPIRES was successfully used in a test demonstration of a personnel file earlier this year. The amount of programming necessary to handle additional attributes or output formats is small relative to the rest of the system. The later version of the system, on which we are now beginning some of the design work, will be somewhat more general as we attempt to accommodate other than bibliographic data in a more general data management system. Nevertheless, even that second version of the system may not be general enough to handle all of the complexities of interactive retrieval and editing of scientific data from large archives of physics data or social science archives of public opinion poll and census data.

This plan for successive iterations as we progress to more complex computer systems is desirable for two reasons. One is that there is much that is ad hoc in the development of computer systems. We have no theory to permit us to predict with certainty the range of modifications necessary when a new complexity or generalization is introduced in one part of the system. In other words, we can't predict which straw will break the proverbial camel's back. In fact we don't even know the weight of some of the straws we are adding. A more important reason for planning successive iterations is that the major unknown is how users will interact with the system. We need to study how users interact, what frustrations they have, what mistakes they make, what features they find useful or not useful, and so on. We are not trying to develop an optimal computer system. Rather, we are trying to optimize an interaction between humans and a computer system. Conseqently, the computer system should not be itself optimized in the usual sense. Instead it has to be adapted to the needs and habits of the users.

Economics

A word about costs of the system may be in order here. It is too early to be able to calculate with much confidence the ultimate operating costs of the kind of system we are developing. It may be that computing costs, particularly the costs of mass storage, will have to come down before such a system as we are developing will be economical to operate. On the other hand, it may be that computer systems, like automobiles, may become an expensive necessity after people learn what difference it makes to have one. I've warned the Provost of this university that our greatest danger to the university budget is not that we might fail, but that we might succeed.

Meanwhile, we are now entering a period of extremely high costs in which we will have the costs of operating an expensive prototype system completely in parallel with existing manual operations at the same time as the costs of continued research and development are expanding. Later, there should be some savings resulting from not having to maintain all of the present manual files in parallel with the computer system and a more efficient computer system than the prototype is likely to be. Also, as the member of users increases the cost per user should come down.

For most users of the system I propose that at least the marginal cost associated with his use of the system be charged directly to the user. This would not be appropriate, of course, for internal use by the library staff itself, or for those early users who are willing to suffer the inconvenience of being guinea pigs for the development group to study. The primary reason for this recommendation that users pay at least marginal costs is not to recover the additional revenue, although that will help. Rather, the reason is to provide the feedback mechanism that will let the operators of the information system know what is most needed by their users. The simple market mechanism of pricing should serve to keep the system in touch with user needs. A secondary benefit would be to avoid frivolous use of a very expensive tool. This proposal, on the surface, appears to run counter to one of the most important educational concepts of the past century, namely the concept of free information or free library service to all who wish to use it. That important principle can be better maintained, not by putting all the costs of the information systems into overhead charges that the users pay indirectly (out of tuition fees, research overhead funds, etc.), but by making sure that all members of the university community are given funds to pay for their use. We already have such a mechanism in the Provost's computer fund at this university. The same or an analogous mechanism could be used to pay for information services. The same money that the university spends on information services anyway can be distributed to users as tokens that can be spent only at the library or information service. Such an apparently radical proposal is sensible for computer information services because of the capability of the computer system to inexpensively maintain the necessary accounting and billing services.

Implementation Progress and Problems

Our development strategy continues to be one of maintaining responsiveness to user needs. In our initial stages we conducted many interviews with high-energy physicists and with some librarians, and performed a secondary analysis of questionnaires from an American Institute of Physics study.

In our prototype system we have not had internal machine efficiency as one of our major goals. Rather we have attempted to develop as quickly as possible a system that potential users can interact with so we can find from a study of their interaction whether we are really building the kind of system that is meeting their needs. We expect to learn enough from the experience of developing the prototype and from how users interact with it, that a second iteration will be necessary in any case. This goal of optimizing a man-machine interaction is a somewhat frustrating one for many good systems programmers. They would like to get on with the job of developing a full-blown system that has an elegant and efficient internal structure. Interrupting their work for frequent 'demonstrations' of partially developed systems and user tests that always result in suggested changes tends to be an unwelcome frustration that they would rather do without. They often tend to feel that they could finish the entire system sooner if people would only leave them alone to get on with the job. The SPIRES project staff have been extremely patient in the face of such frustrations, primarily because they are able to see the logic of adjusting the computer system to meet the needs of the users, and to live with the frequent user interaction that such a premise entails.

These goals dictated our choice of hardware, operating system and programming language. We are utilizing a partition in the 360-67 that does not involve the dynamic relocation hardware specific to that machine. The machine itself, since it is the central computer of the Stanford Computation Center, campus facility, was a logical choice for what was at first a research project and later a prototype development. By staying with a standard language, PL/1, and the 360 Operating System (OS), we will remain compatible with most hardware in the IBM 360 series. The overhead costs associated with such a general operating system and programming language may be more than can be carried for long in a system that must be responsive to cost effectiveness criteria. Meanwhile, we economize on our scarce resource, namely skilled systems programmer time, at the expense of computer time. Nevertheless, the moment of truth must come, and we have still to face the hard decisions about which machine, what operating system, and what programming language for the follow-on system.

The most formidable stumbling block in the way of our development was the need for a suitable time-sharing system to permit multiple users to interact with the same system at what appears to the users to be the same time. When we first started this project we had naively hoped that IBM's TSS (Time Sharing System) would provide a general purpose time-sharing system under which we could operate. Rather than give up in frustration when that didn't materialize, our project staff have designed and programmed a special purpose system. Within the last week we have had successful tests with five users interacting simultaneously and have designed the facilities necessary to expand the number of users up to the current physical capacity of the machine, namely 62 users. There is still more work to be done and undoubtedly there will be more 'bugs' to be tracked down, but we are currently optimistic that the basic system will be up and running by January 1.

File Organization

What is perhaps the key problem in any information retrieval system is the file organization. Given the requirement of rapid retrieval from very large files, a technique of serially searching the file, although useful in batch processing systems, had to be ruled out. Various more or less complicated organizations can be chosen, including threaded lists, directed graphs, balanced tree structures, and others. Some of the considerations to be taken into account include speed of searching, the characteristics of the storage medium or media, ease of making modifications to the file, and the costs of input and file reorganization, if needed. The structure chosen and implemented for SPIRES may not be optimal in some ultimate sense. It does have two important virtues--it is simple, and it does the job. Records are entered into the data collection serially in order of input, with no ordering or organization imposed on them. The costs of periodically reorganizing a Library of Congress file or the holdings file for a large research library would prove too expensive in almost any other organization. The structuring necessary is provided in the index files, which are merely what is called 'inverted files' or inverted lists. We avoid serial searching of index files by using a technique called 'hash coding.'

*Associated with each key in each index (e.g. each author name in the author index) there is a list of the locations of all entries containing that key. Serially searching or chaining through a list of keys in a small segmet of an index file held in the core memory is not a major task for a computer with the speed of the 360/67. The problem is to minimize the number of accesses to the slower disk storage device (in our case a 2314 disk which has a capacity of approximately 208 million characters of information). This is accomplished approximately as follows: The amount of storage required for the index file is divided by the size of segment (or block) that can conveniently be brought into storage in one access to the disk. The result is the number of different blocks in that index. Some of those blocks are reserved as overflow blocks. The rest are labelled primary blocks. We assign each index term to a particular block by taking some part of each search key, for example the first three characters of an author's name, pass the internal computer representation of those characters to a computer routine that interprets it as a number, which is divided by the number of primary blocks. The result of the division is discarded and the remainder gives the

---

*This paragraph was omitted in the oral presentation.

number of the block into which the key is inserted (during the index building operation) or from which it will be retrieved (during the retrieval operation). If the designated primary block is full, then one of the overflow blocks will be linked to the primary block. Thus the appropriate segment of each index file can be searched with usually only a single access to the disk storage. The index files currently implemented for one or more data collections are Author, Title Word, ID Number, Corporate Author, Conference Author, Keyword, Citation (i.e. journal, volume, and page number of journal articles cited in footnote citations or reference lists). Restricting a search on date (i.e. before 1967, after 1965, etc.) is handled in a slightly different way. Each entry in each of the other indexes included not only the location of the document reference, but the date of the document.

Query Language

From the point of view of the user the window into the system or the handle on the bibliographic tool is the query language. This is a particularly critical area in an interactive system in which the ultimate consumers, students, faculty members, their secretaries, library clerical staff, etc. are directly formulating the query without intervention by trained librarians or programmers. The user shouldn't have to know the internal working of the system any more than a housewife driving a late model car with automatic transmission and all the automatic extras needs to be a trained mechanic or automobile manufacturer. But, like the housewife on her way to the grocery store, the computer user has to smoothly and easily control the powerful machine to get where he or she wants to go. Our concept of a good interactive system is not, repeat not, one in which an intelligent computer system analyzes the user's natural language input and decides what the user really wanted. In short, we are not trying to simulate a good reference librarian. Instead we are trying to provide a simple query language in which users can give simple unambiguous instructions that allow them to get the computer to do what they want it to do. Those instructions should be in a language as close to natural

English as possible without introducing ambiguities. The query language we have implemented consists of names of index files that can be searched, followed by the value of what is to be sought in those indexes (e.g. author smith and title library automation). More complex searches can be constructed by combining simple searches with the logical operators "and," "or," and "not." At the end of each input line in the search request, the system replies with the number of documents that have been accumulated using the search specifications. When the number is sufficiently small that the user wishes to see the actual document references, then the command "output" will result in the appropriate information being displayed at the terminal. For those of you who are either computer buffs or linguistics buffs, I can say cryptically that the syntax analyzer we are using employs a simple precedence context-free grammar implemented with a single push-down stack. Allen Veaner said kind things about this language implementation in his talk yesterday, but frankly, we are not satisfied with it. Having implemented a first version with a context-free grammar we are itching to get on to a more sophisticated syntax analyzer which can interpret context. For example we now have to say "Find author smith or author jones" rather than "find author smith or jones" because our syntax analyzer isn't sophisticated enough to look back at the context of the "or" to see that the index file named author was implied. Instead it expects to find the name of an index file after the logical connector and gives the user a frustrating error message, such as "_or_ may not be followed by _jones_."

Input/update.

One important area in any system is how to get the information into the system in the first place, and how to correct it once it gets there incorrectly. We hope that the large majority of our input will come from magnetic tape sources (e.g. LC MARC records) that don't need to be keyboarded locally. If the Library of Congress can't get us bibliographic records fast enough for us to use in our acquisition system and we have to keyboard the information locally in order to produce purchase orders and other output, then the costs of our system will be vastly greater than we would like them to be. I hate to be in the position of being that dependent on other people's efforts, and fervently hope that the Library of Congress will come through with timely data. Delays of merely a few weeks will be very costly to us and, I presume, other users of the MARC tapes. One of our input tapes now is Nuclear Science Abstracts. We hope to expand this kind of service, after we have digested our present commitments and can find funds for the expensive data storage costs, to include magnetic tape outputs from the American Chemical Society Information System, MEDLARS, and other systems.

Meanwhile some data does have to be keyboarded locally. One such collection is the Stanford Linear Accelerator Center preprint collection. We have considered alternate input devices and have settled on on-line input through a time-shared text editing system as the appropriate way for us. Once the documents are correctly keyboarded they are added to the appropriate data collection and appropriate index entries are constructed in a batch job. But the keyboarding itself is done on-line. This has the advantage of letting clerks use an IBM selectric typewriter instead of a keypunch or paper tape machine. Corrections within a line can be made merely by backspacing and striking over. Striking a single key can delete an entire line. If a word is incorrectly spelled more than once a single change command changes all occurrences. The current charges for use of this on-line text editing system are $4 per terminal hour. We were aware that some suspicious reviewers of future proposals might say that this was an impossibly extravagant way to input data so we conducted a cost effectiveness study. From the point of view of SPIRES/BALLOTS I think this will rank as the most cost-effective cost-effectiveness study on record. We got a third party to conduct the study and a fourth party to pay for it. The ERIC clearinghouse located in our Institute for Communications Research, the clearing house for educational media and technology, has been experimenting with use of SPIRES. They agreed to hire Charlie Bourne of Programming Services Incorporated to perform a comparative cost analysis of keypunched input and on-line input. We took a 1,000 document collection and divided it into two sub-collections of 500 documents each for purposes of comparing the two input methods. The results were pleasantly surprising from our point of view. The cost per document was 76.6 cents using the on-line input and $1.331 per document using the keypunch. After correcting for some unexpected computer expenses in processing the cards, the projected future expense for keypunched input was 75 cents per document, still within a penny per document of the on-line input. The major differences were in labor costs, particularly for the cost of corrections. These differences helped to offset the 24 cents per document computer cost associated with using an on-line terminal. The results were instructive to us and might even generalize to other places, for example, any other place where you can buy on-line text editing services for $4 per hour or less.

We expect to complete the programming this fall on the generalized update program that will make it easy to make changes in documents already stored in the computer file, with the appropriate changes in the inverted index files being automatically made. That program will still be a batch program. We felt that attempting an on-line file update at this time was more of a problem that we cared to face. A true on-line update is one of the requirements we have for the next iteration of the system after we have more experience with the prototype version we hope to have operational by January 1. Meanwhile, for the next month or two we will continue to use a rather rudimentary update program that allows us to add and delete entire bibliographic entries. This allows us to make any changes we wish and it does make all the appropriate changes in the indexes, but it is a cumbersome temporary expedient.

Terminals

I have been gratified to see how the eyes of librarians, students, and even faculty colleagues sometimes light up when they sit at a typewriter terminal for a demonstration and see the potential of interactive searching. Nevertheless, I don't believe we can provide a satisfactory system with typewriter terminals alone. The problem with typewriter terminals is that the speed of a typewriter is much slower than human reading speed. I suspect that after the novelty wears off, people will find use of the slow typewriter terminals very frustrating, particularly if they hear that there is a better way to do it. Our original plan was to provide service in IBM 2260 CRT display terminals by January of next year. We did in fact successfully demonstrate search capability from 2260s this past summer, as we had earlier on the much more expensive IBM 2250 CRT display. That experience, plus the fact that better display terminals are soon coming onto the market, led us to cancel our order for 2260s. We are now negotiating for the purchase of a CRT display system with much better characteristics. The key feature of the system is that it uses standard television sets as the display device (although we propose to order a version with a different phosphor and a Polaroid face-plate to reduce the flicker problem. Although the television set is a more complicated device than needed for CRT display, it has the advantage of mass production. It will permit us to display a very readable character set including all the characters in the 96 character ASCII character set in lines of 72 characters long. The device also has a hardware capability for full graphic display although the computer costs in providing a full graphic capability may preclude that application for other than experimental purposes. Since it is compatible with video transmission of camera images we think it keeps open the possibility of computer controlled display of remotely stored microfiche collections. That may be a less expensive solution to the full text problem than we are likely to obtain from digital storage for quite some time. Our target date for implementation is April, although next July may be more realistic given the hardware interface and software systems effort that must go on between now and then.

Other Services.

As you might gather from the way we have been concentrating on system development, we have so far done very little in the way of providing the applications programming for such services as purchase orders, bibliographies, catalog cards, acquisition lists, and other useful output formats. There will still be a lot of work left after we bring up the nucleus of our prototype system. One feature that we do hope to have ready by January will be a generalized personal file capability that will permit any member of the Stanford community to input his personal files into our format and use our retrieval programs for on-line access to his own records. We also have plans for a selective dissemination of information (SDI) system that will work by having users leave standing search requests with the system to be processed against input files when new collections are added (for example, new preprints or the latest issue of Nuclear Science Abstracts). Instead of mailing the results we'll merely store the results in a file they can query from their terminal (or from one of the public terminals on campus).

If we had more time I'd like to lay out the plans we have for the next five years, or even to give my science fiction talk about what we should expect ten years from now. I think it's obvious that we aren't going to run out of interesting work for quite a long time to come. It's too early to tell whether our efforts will be judged a success or failure, but we are certainly having fun trying.

By way of a closing remark I'd like to share with you my homely management philosophy. I try to hire only people who are smarter than I am and who have all the experience and skills that I don't have, even if I have to pay them more than I make. (And we sometimes have to, given the great demand for first rate people in a field exploding as rapidly as computing is.) I try to enthuse them with a vision of what can be done and then delegate to them both the responsibility and the authority necessary to produce it. But they get one additional assignment. They have to teach me as they go along, so I can learn how to create a sophisticated computer system in case I ever have to.

* * * * * * * * * * * *

Discussion

Weisbrod: I hope that you will include in the printed version or whatever it may be, the section you omitted on file organization.

Parker: I will include it in the printed version. Would it be worth confounding the rest of the people if we read it for five minutes now? Or is it too late in the afternoon?

Weisbrod: I read your hieroglyph, but I'd like more detail.

Parker: OK. Let's talk about it separately then.

King: The previous speaker expressed some enthusiasm about the possibility of using PL/1. As a person who is used to it could you evaluate your experience and tell us whether you'd use it again.

Parker: I agree 100% with Tom Burgess when he says it's a language with beautiful specifications. For our purposes, when our criterion was a language that allowed us to do fairly quickly and easily and with a minimum of programmer time what we wanted to do, it just fits the bill. Our hope when we chose that language was that the rate of improvement in the implementation of PL/1 would be faster than it's in fact turned out to be. It still from our point of view, you know, generates too much code and consequent expenses with the cost of core storage, and is, you know, not as efficient object code as we'd like to have, so that's an open question for the follow on system. We wish PL/1 were more efficient in the implementation. We like the language.

Unidentified Voice: You said that you're going to use *Nuclear Science Abstracts* at times I believe. Do you have access to any other externally generated tapes?

Parker: The externally generated tapes that we have on hand at the moment are *Nuclear Science Abstracts*, DESY tapes from the DESY High Energy Physics Lab in Hamburg. The best we've been able to get out of MARC so far is one eighteen document sample tape. We have completed the agreement necessary to get by way of the American Institute of Physics, a file of journal collections, physics journal articles.

We were originally budgeted to be able to bring into the system the Science Citation Index, although the National Science Foundation's expenditure ceilings are probably going to force us to let that go by the board. We would like to expand that to, you know, such things as Medlars, for example, or the output of the American Chemical Society's system, but that's a lot of expensive storage.

Spaulding: My readings of the results in the most recently documented INTREX report, in terms of inputting, comparing paper tape input as opposed to online input, produced almost exactly opposite results. Since Charles Stevens is here, I wonder if he would say that as precisely as it was in the last annual report, because it did hinge on the software-hardware concern, I think it may be a very interesting point.

Stevens: The difference in costs results are tied directly to what Ed Parker said about four dollars per hour. When ours was calculated and reported we were reporting computer costs at $200 per hour. Does that explain the difference?

Shoffner: This $4 per hour is not $4 per hour of central processor time. It's $4 per hour of terminal time.

Stevens: But ours was calculated that way.

Shoffner: For one terminal?

Stevens: No, ours was calculated for the whole machine.

Shoffner: Your online text editor required the whole machine?

Stevens: That's right.

Shoffner: And was only handling only one terminal?

Stevens: Oh no. But we were paying that much. That's our cost.

Shoffner: You mean for a girl on a typewriter, you were charging at two hundred and three dollars an hour. Like WOW! I mean, why did you bother with the calculations? (laughter) Excuse me, that's an aside. Let me take out after this same point, however, because I don't believe Bourne's results, unless there's something here beyond what I understand. If he's comparing like things, then text editing is not a feature of "onlineness," and you can include the same kinds of text editing features in a batch processing system. Now the $4 an hour rate roughly doubles the hourly charge for the person doing the keying, which means that you come out with the same price. Either way you have doubled the production rate, and I don't believe that. Have you investigated this with Mr. Bourne?

Parker: Yes. I've got his detailed report here, and I'd be glad to let you have a look at it later and let you pour over the detailed cost breakdowns.

Unidentified Voice: What terminal do you use?

Parker: We use 2741 terminal, the IBM selectric typewriter terminal.

Fussler: This was also my question, and it seems to me that the results here are clouded as between a typewriter terminal and a key punch, and you don't know to what extent the online edit features contributed to the results.

Parker: Partly, it's the ease of using the selectric typewriter, and partly it's online editing, partly it's the cumbersomeness of the keypunch, and so on. We've compared two things. Both of those things are complex.